目　录

第1章 绪　论

当今是信息的时代，信息的获得、加工、处理和应用等都有了飞跃发展。图像信息处理已经得到一定的发展，但随着对图像处理的要求的不断提高，应用领域的不断扩大，图像理论必须不断补充和发展。图像的处理已经从可见光谱扩展到各个波段，从静止图像发展到运动图像，从物体的外部延伸到物体的内部，以及进行人工智能化的图像处理等。本章主要从介绍数字图像的概念入手，再逐步介绍图像处理的应用领域，以加深读者对图像处理技术的认识。同时介绍了为何要处理数字图像、如何处理，一个标准的图像处理系统是如何组合和运作的等等。

本章主要内容如下：

（1）数字图像概述。

（2）数字图像处理的重要性。

（3）数字图像处理的应用领域。

（4）数字图像处理的基本步骤。

（5）数字图像处理系统。

1.1 引言

视觉是人类最重要的感觉器官。视觉信息是人们由客观世界获得信息的主要来源。人们早已看到，视觉所提供的直观作用远不是语言和文字的描述所能达到的。人类对五彩缤纷的大千世界的认识，大至宇宙星云，小至细胞、基因、分子、原子、病毒，无一不是通过视觉获得深刻的感知而得到的正确的认识。

数字图像处理技术是一门新兴学科，20世纪80年代中期到90年代才形成独立的学科体系。与人类视觉机理的历史相比，它是一门相对年轻的学科。但在其短短的历史当中，它在现实生活中的每一个成功应用，都给我们带来巨大的技术改变。几乎所有与成像相关的领域，数字图像处理都发挥着其巨大的作用。

数字图像处理技术在计算机技术的飞速发展的同时，将不断成熟。目前的计算机设备的价格下降，通过万维网及互联网的网络通信带宽的扩展，图像数字化和图像显示设备的普及，低成本的数字图像处理系统的应用，为数字图像处理技术的持续发展创造了前所未有的机会。数字图像处理技术的另一个巨大的推动力来自不断出现的应用领域，商业、工业、医学、军事、安全系统、计算机视觉和人工智能等等，由此可以预测，数字图像处理在不久的将来会发挥更大的作用。

图像是对客观世界的一种相似性、生动性的描述和写真。图像是客观对象的一种表示，它包含了有关被描述对象的信息。它是人们最主要的信息来源。图像可根据其形式或产生方法来分类。从人眼视觉特点上可将图像分为可见图像和不可见图像。其中可见图像的一个子集为图片，它包括照片、用线条画的图和画；另一个子集为光图像，是用透镜、光栅和信息技术产生的图像。不可见图像包含不可见光成像和不可见量，如温度、压力及人口密度等的分布图。

按波段多少图像可以分为单波段、多波段和超波段图像。单波段图像在每个点只有一个亮度值；多波段图像上每个点不止一个特性。例如：红、绿、蓝三波段光谱图像或彩色图像在每个点都具有红、绿、蓝三个亮度值，这三个值表示在不同光波段上的强度，人眼看来就是不同的颜色。超波段图像每个点具有几十或几百个特性。

按图像空间坐标和明暗程度的连续性可以分为模拟图像和数字图像。模拟图像指空间坐标和明暗程度变化都是连续变化的、计算机无法直接处理的图像。数字图像是一种空间坐标和灰度均不连续的、用离散的数字表示的图像。这样的图像才能被计算机处理，数字图像处理指的就是用计算机对这样的数字图像进行处理，即采用计算机对图像进行信息加工。它与数学、光学、电子学、计算机技术、成像技术、视觉心理学、视觉生理学有关，计算机视觉和人工智能密切相关。如图 1-1 所示，数字图像处理由五块处理功能组成：图像输入、图像存储、图像传送、图像处理、图像显示。

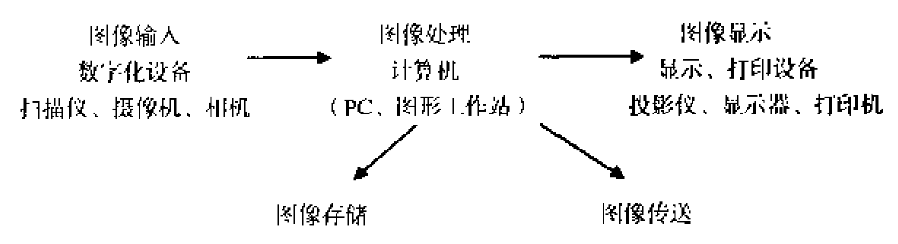


图 1-1　数字图像处理由五块处理功能组成

图像处理的主要内容有：图像的采集、增强、复原、变换、编码、重建、分割、配准、嵌拼、特征提取、模式识别和图像理解等等。本书主要介绍数字图像处理的基础理论：线性系统理论、图像增强、图像复原、图像分析等等。

1.2　数字图像概述

1.2.1　图像和数字图像

从广义上说，图像是自然界景物的客观反映，是人类认识世界和人类本身的重要知识源泉。它是用各种观测系统以不同形式和手段观测客观世界而获得的，可以直接或间接作用于人眼并进而产生视觉的实体。人的视觉系统就是一个观测系统，通过它得到的图像就是客观景物在人眼中形成的影像。图像信息不仅包含光通量分布，而且也还包含人类视觉的主观感受。随着计算机技术的迅速发展，人们还可以人为地创造出色彩斑斓、千姿百态的各种图像。

人们生活在一个信息时代，科学研究和统计表明，人类从外界获得的信息约有 75%来自视觉系统，也就是从图像中获得的。这里图像是比较广义的，包括：照片、绘图、视像等等。图像带有大量的信息，“百闻不如一见”就说明了这个事实。

客观世界在空间上是三维（3-D）的，但一般从客观景物得到的图像是二维（2-D）的。一幅图像可以用一个 2-D 数组 f(x, y)来表示，这里 x 和 y 表示 2-D 空间 XY 中一个坐标点的位置，而 f 则代表图像在点(x, y)的某种性质 F 的数值。例如常用的图像一般是灰度图，这时 f 表示灰度值，它常对应客观景物被观察到的亮度值。

常见图像是连续定义的，即 f、x、y 的值可以是任意实数。为了能用数字计算机对图像进行加工处理，需要把连续的图像在坐标空间 X、Y 和性质空间 F 都进行离散化。这种

离散化了的图像就是数字图像，可以用I(r，c)来表示。这里I代表离散化后的f，(r，c)代表离散化后的(x，y)，其中r代表图像的行（Row），c代表图像的列（Column）。这里I、c、r的值都是整数。本书以后主要讨论数字图像，在不引起混淆的情况下用f(x，y)代表数字图像，如不特别说明，f、x、y都在整数集合中取值。早期英文书籍里一般用picture代表图像，英文picture的原意是指图片、图画、各种照片以及光学影像，是采用绘画或者拍照的方法获得的人、物、景的模拟。现在普遍采用image代表离散化了的数字图像。英文image的含义是"像"，是客观世界通过光学系统产生的视觉印象。图像中每个基本单元叫做图像元素，简称像素（Picture Element）。对2-D图像，英文里常用pixel代表像素。对3-D图像，英文里常用volume element代表其基本单元，简称体素。

与数字图像相关的概念有视频（Video）、图形（Graphics）和动画（Animation）。

视频——视频图像又称为动态图像、活动图像或者运动图像。它是一组图像在时间轴上的有序排列，是二维图像在一维时间轴上构成的序列图像。考虑到人眼的视觉特征，视频图像的刷新速度都有一个明确的限制。如：NTSC制式的电视视频是30帧/秒（fps，frame per second），PAL制式的电视视频是25帧/秒，电影则是24帧/秒。

图形——图形是图像的一种抽象，它反映图像的几何特征，例如：点、线、面等。图形不直接描述图像中的每一点，而是描述产生这些点的过程和方法，被称为矢量图形。矢量图形以解析的形式描述一幅图中所包含的直线、圆、弧线的形状和大小，甚至可用更复杂的形式表示图像中的曲面、光照、材质等。图形的矢量化能够对图中多个部分分别进行控制。所有图形都可用数学的方法加以描述，因而可以对其中任何对像进行任意的变换：放大、缩小、旋转、变形、移位、叠加、扭曲等，但仍保持图形特征。图形变换的灵活性以及处理上的更大自由度等都给计算机图形学的发展带来了巨大的活力。

动画——动画属于动态图像的一种。它与视频的区别在于视频的采集来源于自然的真实图像，而动画则是利用计算机产生出来的图像或图形，是合成的动态图像。动画包括二维动画、三维动画、真实感三维动画等多种形式。

【例1-1】数字图像示例。

图1-2给出两幅典型的数字图像。图（a）所用的坐标系统常在屏幕显示中采用，它的原点O(Origin)在图像的左上角，纵轴标记图像的行，横轴标记图像的列。I(r，c)既可代表这幅图像，也可表示在（r，c）行列交点处的图像值。图（b）所用的坐标系统常在图像计算中采用，它的原点在图像的左下脚，横轴为X轴，纵轴为Y轴。f(x，y)既可代表这幅图像，也可以表示在(x，y)坐标处像素的值。



图1-2 数字图像示例

1.2.2 图像技术和图像工程

图像技术在广义上是各种与图像有关的技术的总称。目前人们主要研究的是数字图像，主要应用的是计算机图像技术。这包括利用计算机和其他电子设备完成的一系列工作，例如：图像的采集、获取、编码、存储和传输，图像的合成和产生，图像的显示和输出，图像的变换、增强、恢复（复原）和重建，图像的分割，目标的检测、表达和描述，特征的提取和测量，序列图像的校正，3-D 景物的重建复原，图像数据库的建立、索引和抽取，图像的分类、表示和识别，图像模型的建立和匹配，图像和场景的解释和理解，以及基于它们的判断决策和行为规划等等。另外，图像技术还可包括为完成上述功能而进行的硬件设计及制作等方面的技术。

尽管计算机图像技术的历史可追溯到 1946 年世界上第一台电子计算机的诞生，但在 20 世纪 50 年代计算机主要还是用于数值计算，满足不了处理大数据量图像的要求。在 60 年代，第三代计算机的研制成功以及快速傅里叶变换算法的发现和应用使得对图像的某些计算得以实现，人们从而逐步开始利用计算机对图像进行加工利用。在 70 年代，图像技术有了长足的进展，而且第一本重要的图像处理专著《Rosenfeld 1976》也得以出版。在 80 年代，各种硬件的发展使得人们不仅能处理 2-D 图像，而且开始处理 3-D 图像。许多能获取 3-D 图像的设备和处理分析 3-D 图像的系统研制成功，图像技术得到了广泛的应用。进入 90 年代，图像技术已逐步涉及人类生活和社会发展的各个方面。以近年得到广为宣传和应用的多媒体技术为例，图像在其中占据了主要地位。广义上来说，文本、图形、视频等都需要借助图像技术才能充分利用。由此，图像技术必将得到进一步的发展和应用，从而改变人们的生活方式。

由于图像技术近年来得到极大的重视和长足的进展，出现了许多新理论、新方法、新算法、新手段、新设备。众所周知，工程是指将自然科学的原理应用到工业部门而形成的各学科的总称。图像工作者普遍认为亟需对图像和它们的处理技术进行综合研究和集成应用，这个工作的框架就形成了图像工程。图像工程学科是将数学、光学等基础科学的原理结合在图像应用中积累的技术经验而发展起来的。

图像工程的内容非常丰富，根据抽象程度和研究方法等的不同可分为三个层次（如图 1-3 所示）：图像处理、图像分析和图像理解。换句话说，图像工程是既有联系又有区别的图像处理、图像分析及图像理解三者的有机结合，另外还包括对它们的工程应用。

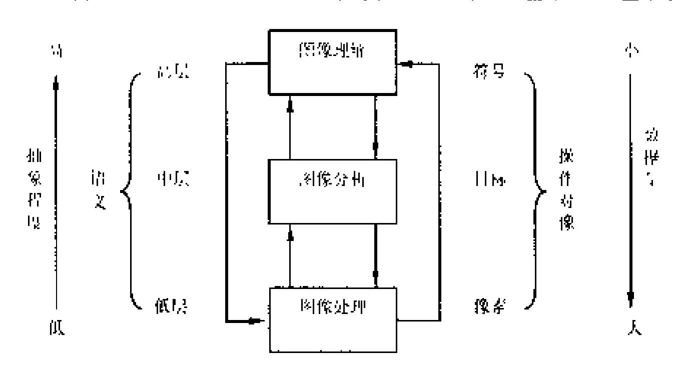


图 1-3　图像工程三层次示意图

图像处理着重强调在图像之间进行的变换。虽然人们常用图像处理泛指各种图像技术，但比较狭义的图像处理主要满足对图像进行各种加工，以改善图像的视觉效果并为自

动识别打下基础；或对图像进行压缩编码，以减少所需存储空间或传输时间、传输通路的要求。

图像分析则主要是对图像中感兴趣的目标进行检测和测量，以获得它们的客观信息，从而建立对图像的描述。如果说图像处理是一个从图像到图像的过程，则图像分析是一个从图像到数据的过程。这里数据可以是对目标特征测量的结果，或是基于测量的符号表示。它们描述了图像中目标的特点和性质。

图像理解的重点是在图像分析的基础上，进一步研究图像中各目标的性质和它们之间的相互联系，并得出对图像内容含义的理解以及对原来客观场景的解释，从而指导和规划行为。如果说图像分析主要是以观察者为中心研究客观世界（主要研究可观察到的事物），那么图像理解在一定程度上是以客观世界为中心，借助知识、经验等来把握整个客观世界（包括没有直接观察到的事物）。

由上所述，图像处理、图像分析和图像理解是处在三个抽象程度和数据量各有特点的不同层次上。图像处理是比较低层的操作，它主要在图像像素级上进行处理，处理的数据量非常大。图像分析则进入了中层，分割和特征提取把原来以像素描述的图像转变成比较简洁的非图形式的描述。图像理解主要是高层操作，基本上是对从描述抽象出来的符号进行运算，其处理过程和方法与人类的思维推理可以有许多类似之处。另外，由图 1-3 可见，随着抽象程度的提高，数据量是逐渐减少的。具体说来，原始图像数据经过一系列的处理过程逐步转化为更有组织和用途的信息。在这个过程中，语义不断引入，操作对象发生变化，数据量得到了压缩。另一方面，高层操作对低层操作有指导作用，能提高低层操作的效能。

1.3 数字图像处理的重要性

图像是人们从客观世界取得信息的重要来源。

人类传递信息的主要媒介是语音和图像。在接受的信息中，听觉信息占 20%，视觉信息占 60%，其他如味觉、触觉、嗅觉总的加起来不过占 20%。图像信息处理是人类视觉延续的重要手段。人的眼睛只能看到可见光部分，波长为 380nm～780nm，而迄今为止人类发现可成像的射线已有多种，如：γ射线、X 射线、紫外线、红外线、微波，它们扩大了人类认识客观世界的能力。

数字图像处理是一个跨学科的前沿科技领域。在诸如工程学、计算机科学、信息学、统计学、物理、化学、遥感、生物医学、地质、海洋、气象、农业、冶金等许多学科中都存在一个重要的课题：怎么获取图像？如何使用获得的图像满足人的视觉理解的要求？为了提高存储和传输的效率，如何进行编码和压缩？如何分析和理解一幅图像？

数字图像处理技术产生的经济效益显著。图像处理技术发展到今天，许多技术已日趋成熟，在各个领域的应用取得了巨大的成功和显著的经济效益。例如：CT、复印、传真、指纹识别等均受益于图像处理技术发展。

1.4 数字图像处理的应用领域

计算机图像处理和计算机、多媒体、智能机器人、专家系统等技术的发展密切相关。近年来计算机识别、理解图像的技术发展很快，图像处理除了直接供人观看外，还发展了

与计算机视觉有关的应用，如：邮件自动分拣、车辆自动驾驶等。下面仅列举一些典型应用实例，实际应用远远超出以下介绍的内容。

1. 生物医学中的应用

（1）显微图像处理（见例 1-2）。

（2）DNA（脱氧核糖核酸）显示分析。

（3）红、白血球分析计数。

（4）虫卵及组织切片的分析。

（5）癌细胞识别。

（6）染色体分析（见例 1-3）。

（7）DSA（心血管数字减影）及其他减影技术。

（8）内脏大小形状及异常检查。

（9）微循环的分析判断。

（10）心脏活动的动态分析。

（11）热像分析，红外像分析。

（12）X 光照片增强、冻结及伪彩色增强。

（13）超声图像成像、冻结、增强及伪彩色处理。

（14）CT（见例 1-4），MRI，γ射线照相机，正电子和质子 CT 的应用。

（15）专家系统如手术 PLANNING 规划的应用。

（16）生物进化的图像分析。

数字图像处理技术目前在医学领域的应用是比较广泛和比较成熟的，例 1-2 是一个细菌的显微图像的几个数字处理方法和二值化处理的过程。例 1-3 是一个医学上利用图像处理技术处理后的人脑的 CT 图像阵列显示，例 1-4 是人体血液细胞的数字图像显示，例中的图像基本上是经过图像增强技术处理之后的效果。

【例 1-2】如图 1-4 所示，在显微照相技术得到的图片的数字处理过程，这是一个很简单但是很常用的图像处理方法，通过图像的二值化处理，找到目标物体的边缘。这种边缘提取的方法比较简单，也比较常用，主要用在目标物体和背景的灰度差别较大，并且灰度直方图的双峰中的峰谷比较明显，比较好确定阈值。



（a）显微图像获得的微生物灰度图







（b）图像二值化分离出目标物体　（c）获取目标物体边缘

图 1-4 显微图像处理实例

【例1-3】如图1-5所示，数字图像处理技术应用于医学图片的显示。

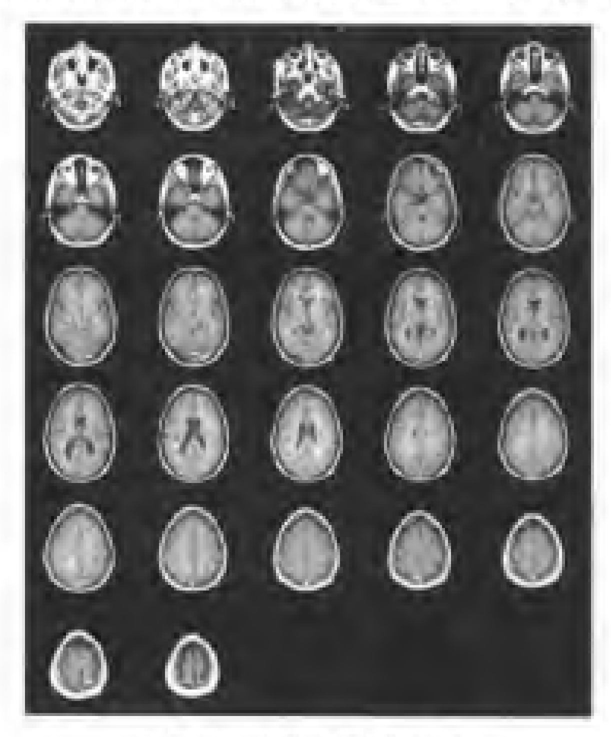


图1-5 人脑CT图片显示

【例1-4】如图1-6所示，血液细胞的图像显示。

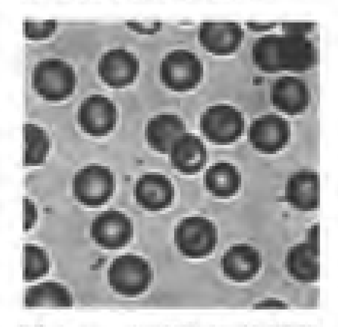


图1-6 血液细胞的图像

2. 遥感航天中的应用

（1）军事侦察、定位、引导、指挥等应用。

（2）多光谱卫星图像分析。

（3）地形、地图、国土普查。

（4）地质、矿藏勘探。

（5）森林资源探查、分类、防火。

（6）水利资源探查，洪水泛滥监测。

（7）海洋、渔业方面如温度、渔群的监测预报。

（8）农业方面如谷物估产、病虫害调查。

（9）自然灾害、环境污染的监测。

（10）气象、天气预报图的合成分析预报。

（11）天文、太空星体的探测及分析。

（12）交通、空中管理、铁路选线等。

【例 1-5】通常通过卫星所拍摄的地面图片，由于大气层的干扰，往往会有周期性噪声的影响，直接得到的图片效果非常差，而通过数字图像处理技术的应用，基本上可以克服这个问题，去噪也是图像复原技术的一个重要手段，图 1-7 是一幅由高空向地面拍摄的地面图像（处理后的图像）。



图 1-7 从高空向地面拍摄的地面图像

3. 工业应用

（1）CAD 和 CAM 技术用于模具、零件制造、服装、印染业。

（2）零件、产品无损检测，焊缝及内部缺陷检查。

（3）流水线零件自动检测识别（供装配流水线用）。

（4）邮件自动分拣、包裹分拣识别。

（5）印制板质量、缺陷的检查。

（6）生产过程的监控。

（7）交通管制、机场监控。

（8）纺织物花型、图案设计。

（9）密封元器件内部质量检查。

（10）光弹性场分析。

（11）标识、符号识别如超级市场算账、火车车皮识别。

（12）支票、签名、文件识别及辨伪。

（13）运动车、船的视觉反馈控制。

4. 军事、公安领域中的应用

（1）巡航导弹地形识别。

（2）指纹自动识别。

（3）罪犯脸形的合成。

（4）雷达地形侦察。

（5）遥控飞行器的引导。

（6）目标的识别与制导。

（7）警戒系统及自动火炮控制。

（8）反伪装侦察。

（9）手迹、人像、印章的鉴定识别。

（10）过期档案文字的复原。

（11）集装箱的不开箱检查。

5. 其他方面的应用

（1）图像的远距通信。

（2）多媒体计算机系统及应用。

（3）电视电话。

（4）服装试穿显示。

（5）理发发型预测显示。

（6）电视会议。

（7）办公自动化、现场视频管理。

（8）文字、图像电视广播。

1.5 数字图像处理的基本步骤

后面各章涉及到的图像处理可划分为两个主要类别：一类是输入与输出都是数字图像；另一类，输入的是图像，输出的是从图像中提取的属性特征。图 1-8 给出了数字图像处理的基本步骤。

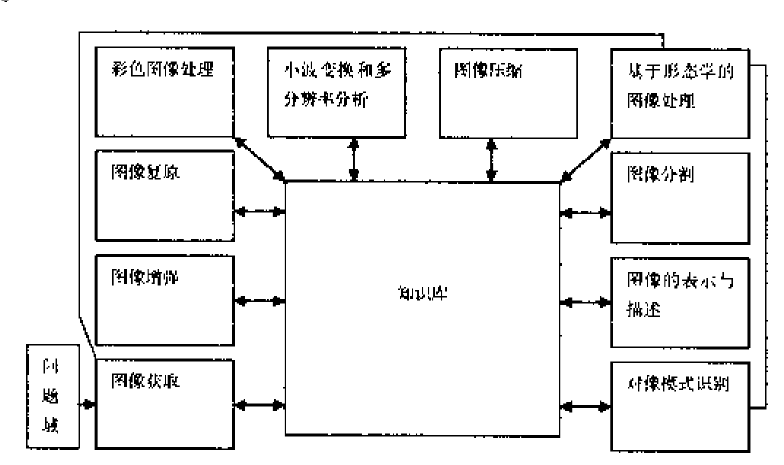


图 1-8 数字图像处理的基本步骤

图像获取是图 1-8 中的第一步处理。在这一章，要介绍一些基本的数字图像概念，这些概念将贯穿本书。获取一幅数字形式的图像与给出一幅数字形式的图像一样简单。通常，图像的获取包括如何设置比例尺等预处理。

图像增强是数字图像处理领域最简单又最有吸引力的领域。简单来说，图像增强技术的基本思路就是显现那些被模糊了的细节，或简单地突出一幅图像中某些特征。图像增强是图像处理中非常主观的领域，这一原则十分重要。本书并没有专门设立一章来讨论图像增强技术，而是在第 9 章的图像的灰度直方图及基本运算以及第 7 章的滤波器设计中涉及了图像增强技术。因为数字图像处理的数学以及其他学科的知识要求比较高，而图像增强的技术知识来源比较广泛，涉及很多其他的知识，所以本书中先对基本的准备知识，如线性系统、傅立叶变换和小波分析做系统的介绍。

图像复原也是改进图像外貌的一个重要处理领域。它和图像增强的根本不同点在于图像复原是客观的。在某种意义上，复原技术倾向于以图像退化的数学或概率模型为基础。相反，增强是以如何构成好的增强效果这种人为的主观判断为基础。

彩色图像处理目前已经成为一个越来越重要的领域。由于彩色图像处理技术和灰度图的数字图像处理技术有很强的关联性，所以本书主要介绍灰度图的数字图像处理，在第13章介绍了彩色模型、彩色图像的噪声等基本概念。

小波分析是在各种分辨率下描述图像的基础，多分辨率图像处理也是在这个基础上发展起来的，近年来小波的应用十分广泛，成为数字图像处理中一个最重要的技术，本书对小波变换的数学基础做了一个比较详细的介绍，这些理论也被用于图像处理数据压缩金字塔描述方法，在算法中图像被成功的细分为较小的区域。

图像压缩，所涉及的技术是减少图像在计算机中的存储量，或者在图像传输时降低频带。虽然存储技术在过去的十年内有了很大的进步，但对于传输能力还没有找到突破性的办法，尤其在互联网上更是如此，互联网以大量的图片内容为特征，目前彩屏手机的兴起，对图像传输技术又有了更高的要求。图像压缩技术对应的图像文件扩展名对大多数计算机用户是不陌生的，例如JPG文件扩展名用于JEPG（国际联合图片专家组）图像压缩标准，在第2章的数字图像基础中会对图像文件格式做一个详细的介绍。

基于形态学的图像处理涉及到提取图像的工具，它在图像的表示与描述方面是非常有用的。在图像分割中，基于形态学分水岭的分割就是一个很好的应用。

图像分割就是将一幅图像划分为背景和目标物，其中目标物是人们感兴趣的部分，其他的称为背景。通常情况下，自主分割是数字图像处理中最困难的任务之一。要想成功地将物体分别识别出来（即首先与背景分离），需要十分复杂的分割过程和大量的处理工作，包括图像的预处理。通常不稳定的分割算法几乎最终都会导致失败。一般情况下，分割越准确，识别越成功。

图像的表示与描述总是在图像分割步骤的输出后面，通常这个输出是未加工的数据，它的构成或者是区域的边缘（区分不同图像区域的像素集合），或者是区域本身所有的点。所以把数据转换成适合计算机处理的形式是非常必要的。首先，要确定数据是被表观为边界还是整个区域。若关注的是物体的外部形状特征，则边界表示是合理的。相反若关注的是内部特征，则区域表示比较合适。通常在应用中它们是互补的。当然，为了使人们感兴趣的特征更加明显，应使用描述的方法，也叫做特征提取，提取的特征是人们感兴趣的某些信息或区分自己和其他信息的基本特征。

最后的步骤是对象模式识别，如果把识别对象看作最终目的，那前面的所有图像处理的步骤可以看作是对象的图像显示的预处理。目前的识别领域飞速发展，例如指纹识别、语音识别、虹膜识别、人脸识别、耳轮识别和步态识别等等，都对数字图像处理技术有着很高的要求，这些领域的不断发展，同时也促使图像处理技术的不断创新与发展。

本书中虽然并没有详细讨论图像显示，但是图像显示也是非常重要的。图像处理的结果的观察可在前页图1-8中任何一级的输出进行。还要强调的是，并不是所有的图像处理应用过程都需要前页图1-8中那样复杂的交互，实际上，如果图像处理任务不太复杂，只需要针对需要进行有关处理步骤即可。

1.6 数字图像处理系统

1.6.1 图像处理和分析系统

一个基本的图像（处理和分析）系统构成可由图 1-9 表示。图中各模块都有特定的功能，分别是采集、显示、存储、通信、处理和分析。为完成各自的功能每个模块都需一些特定的设备。图像采集可采用电荷耦合器件（Charge Coupled Devices，CCD）的照相机、带有视像管（Vidicon）的视频摄像机和扫描仪（Scanners）等。图像显示可用电视显示器（TV Monitors）、随机读取阴极射线管（Cathode Ray Tubes，CRT）和各种打印机（Printers）等。图像存储可采用磁带（Magnetic Tape）、磁盘（Magnetic Disks）、光盘（Optical Disks）和磁光盘（Magneto-Optical Disks）等。图像通信可借助综合业务网（ISDN）、计算机局域网（LAN），甚至普通电话网（PSTN）等。最后，图像处理和分析主要是运算，所使用的设备主要是计算机，当然必要时还可借助专用硬件。

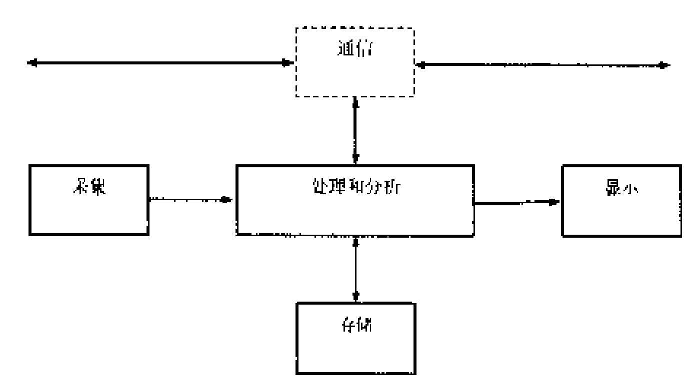


图 1-9 图像系统的构成示意图

1.6.2 图像的采集

为采集数字图像，需要两种装置（器件）。一种是对某个电磁能量谱波段（如 X 射线、紫外线、可见光、红外线等）敏感的物理器件，它能产生与所接受到的电磁能量成正比的（模拟）电信号；另一种称为数字化器，它能将上述（模拟）电信号转化为数字（离散）的形式。所有采集数字图像的设备都需要这两种装置。

以常见的 X 光透视成像仪为例，由 X 光源发出的射线穿越物体到达另一端对 X 光敏感的媒体。这个媒体能获得物体对 X 光不同吸收率的图像。它可以是胶片，一个带有将 X 光转化为光子的电视摄影机，或其他能将 X 光转换为数字图像的离散检测器。

用于可见光和红外线成像的设备主要有显微密度计（Micro-Densitometers）、析像管（Image Dissector）、视像管和对光子敏感的固态阵列（Solid-Slate Arrays）等。使用显微密度计时，需要数字化的图像应是透明底片或照片的形式。视像管和对光子敏感的固态阵列除可接受以上形式的图像外，还可以将有足够光强入射检测器的自然图像数字化。

在使用显微密度计时，需要将透明底片或照片放在一个平板上或卷在一个圆鼓上。当光线聚焦在图像上时，平移平板或转动圆鼓就可以完成扫描。如果是透明底片，光穿过透明底片；如果是照片，光从表面反射。在这两种情况下，光束都聚焦在光子检测器上，各个检测器根据光强度记录下对应当前位置的图像灰度值。如果灰度值和位置坐标都取整数，

就得到一幅数字图像。尽管显微密度计速度比较慢，但由于机械平移过程的连续本质，它的空间精确度很高。

视像管摄影机的工作原理基于光导性质。聚焦在视像管表面的图像能形成与光学图像的灰度分布相对应的光导模式。用另一个独立的电子束扫描光导管的另一面，由于中和作用，这个电子束在一个接受器上产生与输入光亮度模式对应的压差信号。如果量化这个信号并记录下对应的扫描束位置，就可得到一幅数字图像。

固态阵列是由称为感光基元（Photosites）的离散硅成像元素构成的。这样的感光基元能产生与所接受的输入光强成正比的输出电压。固态阵列可按几何组织形式分为两种：线扫描器和平面扫描器。线扫描传感器包括一行感光基元，它靠场景和检测器之间的相对运动来获得 2-D 图像。平面扫描传感器由排成方阵的感光基元组成，可直接得到 2-D 图像。固态平面传感器阵列的一个显著特点是它具有非常快的快门速度（可达 10^{-4}s），所以能将许多运动瞬间冻结。

固态阵列中主要元件是电荷耦合器件 CCD。图 1-10 给出一个线扫描 CCD 传感器的示意图。这个传感器由一行感光基元，两个定时的将感光基元中的内容传给传输寄存器的传输门，以及一个定时的将传输寄存器中的内容传给放大器的输出门构成。放大器输出的电压信号与感光基元行的内容成比例。

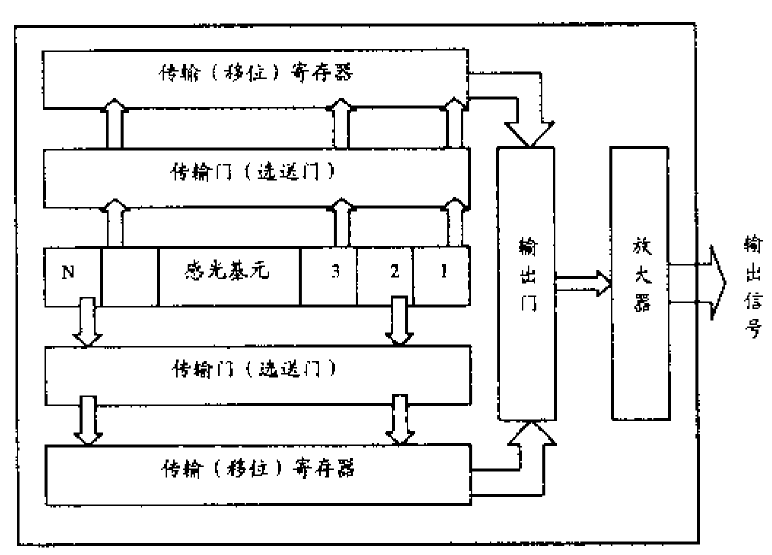


图 1-10　线扫描传感器

电荷耦合平面阵列的工作原理与线阵相似，但感光基元排列成一个矩阵形式并由传输门和传输寄存器隔开。先将奇数列感光基元的内容顺序送进垂直传输寄存器，然后再送进水平传输寄存器。把水平传输寄存器的内容送进放大器就得到 1 帧隔行的视频信号。对偶数列感光基元重复以上过程就可得到另 1 帧隔行的视频信号。将 2 帧合起来就得到隔行扫描电视的 1 场（f）。NTSC 制视频信号的扫描速度是 30f/s，PAL 视频信号的扫描速度是 25f/s。

现在常用的线扫描 CCD 一般有 512 到 4096 个像素或更多，而 4096×4096 个像素的平面扫描 CCD 也已在使用。利用图像处理和分析的手段，还可以通过图像的拼接用较小分辨率的 CCD 获得较大视场的图像。电视摄像机一般用 CCD 阵组成，要得到数字图像需把摄像机的视频输出送到一个数字化器（称为图像捕获器）中。这个工作常通过在计算机中插入专门的硬件捕获卡实现。

1.6.3 图像的显示

对图像处理来说，处理的结果主要用于显示给人看。对图像分析来说，分析的结果也可以借助计算机图形学技术转换为图像形式直观地展示。所以图像显示对图像处理和分析系统来说是非常重要的。

常用图像处理和分析系统的主要显示设备是电视显示器。输入显示器的图像也可以通过硬拷贝转换到幻灯片、照片或透明胶片上。除了电视显示器，可以随机存取的阴极射线管（CRT）和各种打印设备也可用于图像输出和显示。

在CRT中，电子枪束的水平垂直位置可由计算机控制。在每个偏转位置，电子枪束的强度是用电压来调制的。每个点的电压都与该点所对应的灰度值成正比，这样灰度图就转化为光亮度变化的模式，这个模式被记录在阴极射线管的屏幕上。

打印设备一般用于输出较低分辨率的图像。在纸上打印灰度图像的一种简便方法是利用标准行打印机的重复打印能力。输出图像上任一点的灰度值可由该点打印的字符数量和密度来控制。近年来使用的各种热敏、喷墨和激光打印机等具有更高的能力，已可打印较高分辨率的图像。

1.6.4 图像的存储

图像包含有大量的信息，因而存储图像也需要大量的空间。在图像处理和分析系统中，大容量和快速的图像存储器是必不可少的。在计算机中，图像数据最小的量度单位是比特（bit）。存储器的存储量常用字节（1 byte = 8 bit）、千字节（k byte）、兆（10^6）字节（M byte）、吉（10^9）字节（G byte）、太（10^{12}）字节（T byte）等表示。存储1幅1024 × 1024的8bit图像就需要1M byte的存储器。用于图像处理和分析的数字存储器可分为三类：

（1）处理和分析过程中使用的快速存储器。

（2）用于比较快地重新调用的在线或联机存储器。

（3）不经常使用的数据库（档案库）存储器。

计算机内存就是一种提供快速存储功能的存储器。目前，一般微型计算机的内存常为几十M byte。另一种提供快速存储功能的存储器是特制的硬件卡，也叫帧缓存。它常可存储多幅图像并可以视频速度（每秒25或30幅图像）读取。它也可以允许对图像进行放大缩小，以及垂直翻转和水平翻转。目前常用的帧缓存容量在几十M byte到上百M byte。

磁盘是比较通用的在线存储器，常用的Winchester磁盘已可存储多个G byte的数据。近年还常用磁光（Magneto-Optical, MO）存储器，它可在5.25英寸的光片上存储上G byte的数据。在线存储器的一个特点是需要经常读取数据，所以一般不采用磁带一类的顺序介质。对更大的存储要求，还可以使用光盘塔，一个光盘塔可放几十个到几百个光盘，利用机械装置插入或从光盘驱动器中抽取光盘。

数据库存储器的特点是要求非常大的容量，但对数据的读取不太频繁。这里常用磁带和光盘。长13英尺的磁带可存储达G byte的数据。但磁带的储藏寿命较短，在控制很好的环境中也只有7年。一般常用的一次写多次读（Write-Once-Read-Many，WORM）光盘可在12英寸的光盘上存储6G byte数据，在14英寸的光盘上存储10G byte数据。另外WORM光盘在一般环境下可储藏30年以上。在主要是读取的应用中，也可将WORM光

盘放在光盘塔中。一个存储量达到 T byte 级的 WORM 光盘可存储上百万幅 1024×1024 的 8bit 图像。

1.6.5 图像通信

近年来随着信息高速公路的建设，各种网络的发展非常迅速。因而，图像的通信传输也得到了极大的关注。另一方面，图像传输可使不同的系统共享图像数据资源，也极大地推动了图像在各个领域的广泛应用。图像通信可分成近程的和远程的，它们的发展情况不太一致。

近程图像通信主要指在不同设备间交换图像数据，现已有许多用于局域通信的软件和硬件以及各种标准协议。

远程图像通信主要指在图像系统间传输图像。长距离图像通信遇到的首要问题是图像数据量大而传输通道比较窄。例如，目前常用的电话线的速率为 9600 bit/s，如果以这样的速率传输 1 幅 512×512 的 8 bit 图像就需要 300s。利用中继站的无线传输速率比较高，但费用增加得更高。解决这个问题需要对图像数据进行压缩，这要靠下而介绍的图像处理和分析模块完成。

1.6.6 图像处理和分析

对图像的处理和分析一般可用算法的形式描述，而大多数的算法可用软件实现，只有在为了提高速度或克服通用计算机限制的情况下才使用特制的硬件。进入 20 世纪 90 年代，人们设计了各种与工业标准总线兼容的可以插入微机或工作站的图像卡。这不仅减少了成本，也促进了图像处理和分析专用软件的发展。这些图像卡包括用于图像数字化和临时存储的图像采集卡，用于以视频速度进行算术和逻辑运算的算术逻辑单元，以及前而提到的帧缓存。此外，许多基本的图像处理和分析软件包也已商品化。

图像处理和分析中的一个重要事实是对特殊的问题需要特殊的解决方法。现有的图像处理和分析软件和硬件可以提供比以前更多更快的通用工具，但要解决具体的问题还需要深入的研究和开发。本书主要通过对有关基础理论和实用技术的介绍，帮助读者打下进一步研究和开发的基础。

小结

本章主要对数字图像处理的起源、重要性和应用领域做了一个概括性的介绍，主要使读者对数字图像处理的技术概念和应用范围有一个总体的认识。在后而的章节中，将对图像处理的理论基础和应用方而做详细的阐述，并通过系统的理论知识和大量的实例进一步讲解这些技术的应用，以便读者能对数字图像处理有一个全而的了解。

重点：本章的重点在于介绍什么是数字图像，图像技术和图像工程有哪些，数字图像处理的应用领域，图像处理的系统分析。

难点：本章的难点在于如何真正了解数字图像的概念，这样才能更好地掌握数字图像处理技术。

疑点：本章的疑点在于图像处理和分析中的一个重要事实是对特殊的问题需要特殊的解决方法。现有的图像处理和分析软件和硬件可以提供比以前更多更快的通用工具，但要

解决具体的问题还需要深入的研究和开发。本书主要通过对有关基础理论和实用技术的介绍，帮助读者打下进一步研究和开发的基础。

练习一

一、选择题

1. 人的眼睛只能看到的可见光部分，波长约为（ ）。

   A. 380nm ~ 780nm　　B. 38nm ~ 78nm

   C. 3800nm ~ 7800nm　　D. 130nm ~ 380nm

2. 存储1幅1024×1024的8bit图像就需要（ ）的存储器。

   A. 10M byte　　B. 1M byte

   C. 0.1M byte　　D. 1024M byte

二、思考题

1. 连续图像 f(x, y)与数字图像 I(r, c)中各量的含义各是什么？它们有什么联系和区别？它们的取值各在什么范围？

2. 图像处理、图像分析和图像理解各有什么特点？它们之间有哪些联系和区别？

三、上机题

试着编程读取一幅图像并显示。

第 2 章　数字图像基础

本章的目的是介绍一些与图像和视觉有关的基本概念和基本定义，它们大致分成三组：一组是与人的视觉有关的一些知识，包括人眼成像的结构和机理及眼睛适应亮度变化和区分不同亮度的能力，另外还包括颜色视觉以及颜色模型；另一组是有关图像获取的一些内容，包括一种适用于一般视觉过程的基于照明和反射特性的图像（成像）模型，图像采集时的几何投影变换关系，以及图像空间采样和灰度量化的概念；还有一组是有关数字图像的一些基础，包括各像素间的联系（例如连通性和距离测量），以及图像间的简单运算和空间坐标变换。本章主要内容包括：

（1）人眼视觉和色彩空间。

（2）数字图像文件的存储格式。

（3）数字图像表示。

（4）空间和灰度级分辨率。

（5）放大和收缩数字图像。

（6）像素间的基本关系。

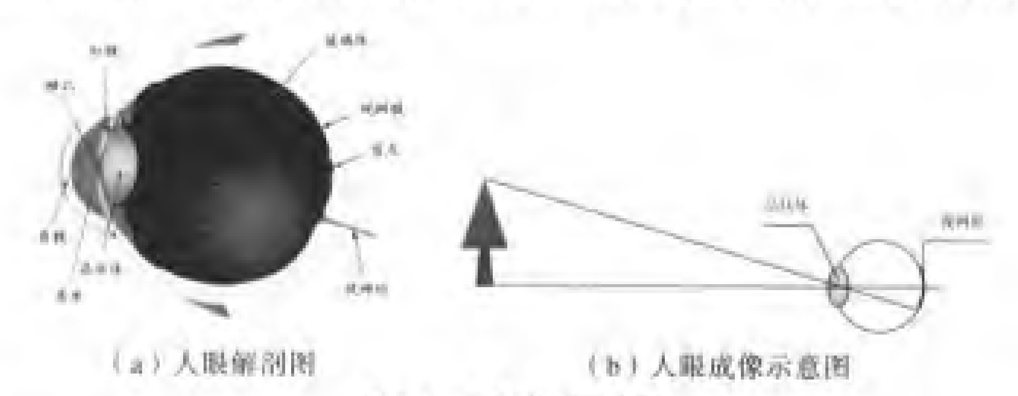
（7）算术和逻辑运算。

（8）坐标变换。

2.1　人眼视觉和色彩空间

2.1.1　人眼成像

人眼的视觉系统是由眼睛和视觉神经系统构成的。人眼的结构如图 2-1 所示。



（a）人眼解剖图　　　（b）人眼成像示意图

图 2-1　人眼截面示意图

眼球被直径约 24mm 的球状巩膜所包围着。巩膜的作用相当于一个保护壳和球状暗箱。球状暗箱的中间开口处是晶状体。晶状体相当于照相机镜头上的凸透镜片。晶状体可以在肌肉的作用下改变形状和厚度。这样相当于照相机镜头的凸透镜的曲率发生变化，从而可以调整焦距，以使被注视物体刚好落在视网膜的表面上，使成像最清晰。熟悉照相的读者都知道照远景要用长焦距镜头（例如 200mm 的镜头），照中景要用中焦距的镜头（例如 70mm 镜头），照近景要用短焦距镜头（例如 35mm 镜头）。但由于一种镜头的焦距是固定的，所

以除了靠换镜头进行粗调整外，还要通过拉伸镜头与底片的距离进行微调。人的眼睛要比照相机科学多了，只是靠直接调整透镜的曲率来调整焦距。当然，如果一个人的眼睛调整机能低下或失调，人就要佩戴眼镜来校正。

晶状体——镜头（Lens）

视网膜（Retina）——成像面

屈光能力——聚焦控制

晶状体前面是虹膜。虹膜形成瞳孔，起到照相机光圈的作用。它可以根据外界光线的强度来调整暗箱口开放的大小，使进入视网膜的光线的量产生的刺激不弱也不强。眼睛的底部是视网膜。视网膜的作用是将光信号变换、滤波和编码程神经系统的内部表达信号（电信号）传给视觉神经系统和中枢神经系统。

眼睛的构造除了新陈代谢需要的一些功能外，其余部分都可以与现在所使用的照相机中的各个功能部件找到一一对应的关系。但是，人类眼睛比照相机更灵活的地方是，照相机在对准目标时要靠人工调节或机械驱动装置辅助人工来实现，而眼睛是靠眼球外侧的六块肌肉的运动来自动调整。这些肌肉移动眼睛使两眼睛的思想只想某一特定位置，并产生双目的视差以利于深度知觉。值得一提的是，现在市场上一些高级的照相机已经有了自动调节焦距的功能，这实际上可以说人类在人工调焦这个问题上已经找到了合适的模型或者脉冲响应函数。

图像是光辐射所产生的能量分布图，通过光波映入眼中产生视觉。光是电磁波的一种形式，频率为105~1025Hz。可见光是其中很小部分，波长为780nm的红光到波长为380nm的紫光。彩色视觉属人眼的一种明视觉。人眼的视网膜由感光细胞覆盖。锥状细胞将电磁光谱的可见部分分为三个波段：红、绿和蓝，人类视觉为三色的。由于人眼对这三种颜色最敏感，因此红、绿和蓝被选为三基色，组成RGB三维色彩空间。

2.1.2　亮度适应和区分

人眼的亮度适应范围10^{10}量级，同时亮度适应区间不大，一般<64级。基于人眼成像的原理，人眼的感觉光亮度并不等于光的强度。但人眼对光的强度变化敏感程度非常强，反而对颜色的变化的敏感程度弱，所以在目前的视频压缩技术中利用这个原理进行有损压缩，压缩效果很好，已经成为网上视频的通用压缩格式，例如.rm的视频文件就是应用这种压缩技术，如图2-2所示。

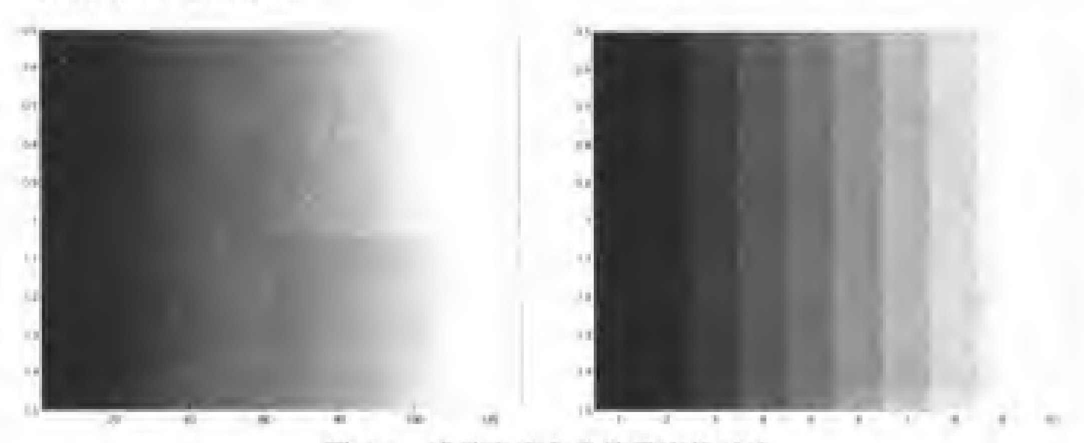


图2-2　感觉亮度和光的强度的对比

【例 2-1】MATLAB 中调用命令（以下类似）：

```
colormap(gray);
imagesc(1:10)
figure(2)
colormap(gray);
imagesc(1:128)
```

【例 2-2】马赫带效应。

如图 2-3 所示，各灰度带之间似乎都有一条明显的分界线。

```
imagesc(1:8);
colormap(gray);
```

【例 2-3】视错觉。

视觉不仅仅是视感觉器官 - 眼睛对外界的反应，同时也是一种心理过程。视觉感知的结果不仅仅取决于刺激本身，同时还受经验、对比等的影响，这就是视觉的相对性。古人曾经争论过"初升的旭日和憎恶的骄阳究竟何者离我们更近"，认为旭日近者的理由是由于旭日看起来更大，认为正午太阳更近者认为正午太阳更烈，所以更近。这实际上就是视觉相对性或者说是知觉相对性产生的结论。典型的视觉相对性有以下两种：

（1）前景与背景。

在视觉中前景是指视觉关心的主体，而背景则是与前景相关联的的其他刺激。前景是主题，背景则是衬托。一幅图像如果没有明显的关心对象，也就是没有前景而只有背景，这样往往会使人对此产生很弱的视觉感知。只有形成前景与背景的强烈对比，特别是当观察者对前景比较熟悉时，才能给人以明显的刺激。这种视觉现象在美术、建筑等方面得到了大量的应用。在计算机数字图像处理技术中，这种关系的研究对于如何提供性能优良的分割算法具有指导性作用。

（2）视知觉的对比。

这种对比是指两种有相对性质的视觉刺激同时出现或相继出现时，由于彼此的影响造成明显差异的现象。这种对比经常在两种情况下发生：一种是前景和背景可以相互置换，例如加拿大国旗上的枫叶，另一种是前景受背景影响，如图 2-4 所示，其实两幅图中心的灰色正方形的颜色与灰度都是一样的，但由于在不同的反差很大的背景下，造成不同的错觉，亮背景下显得暗，暗背景下显得亮。在人类的视觉中尽管这种对比有时会造成与实际情况不符，但却起着重要的作用。

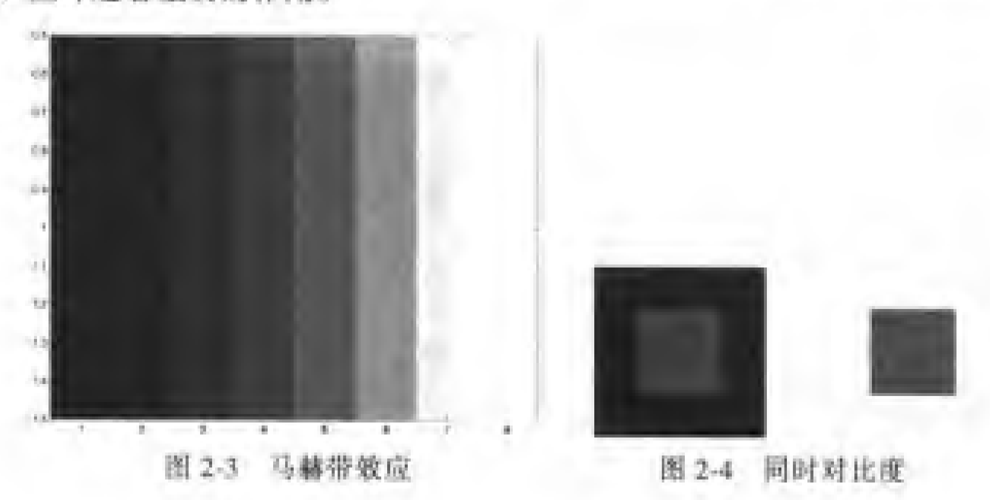


图 2-3　马赫带效应　　　　图 2-4　同时对比度

图例运行程序如下：

```
colormap(gray);
dark = zeros(256, 256);
dark(64:192, 64:192) = 0.5;
subplot(1,2,1);
imshow(dark)
bright = ones(256, 256);
bright(64:192, 64:192) = 0.5;
subplot(1,2,2);
imshow(bright)
```

2.1.3 颜色基础

在图像处理中运用颜色受两个主观因素推动。第一，颜色是一个强有力的描绘子，它常常可简化目标物体的区分以及从场景中提取目标；第二，人可以辨别几千种颜色、色调和亮度，相形之下只能辨别几十种灰度层次。第二个因素对于人工图像分析特别重要。

虽然人的大脑的感知和热物理颜色所遵循的过程是一种生理和心理现象，这一现象还远没有完全了解，但颜色的物理性质可以由实验和理论结果支持的基本形式来表示。在1666年，艾萨克·牛顿（Isaac · Newton）发现一个现象，当一束太阳光通过一个玻璃棱镜时，出现的光束不是白色的，而是由一端为紫色到另一端为红色的连续的彩色谱组成。彩色谱可分为六个宽区域：紫色、蓝色、绿色、黄色、橘红色和红色。当观察全彩色时，色谱末尾的颜色不是突变的，而是一种颜色连续地平滑过渡到下一种颜色。

将颜色分解为红、绿、蓝三基色，色度图如图2-5所示。

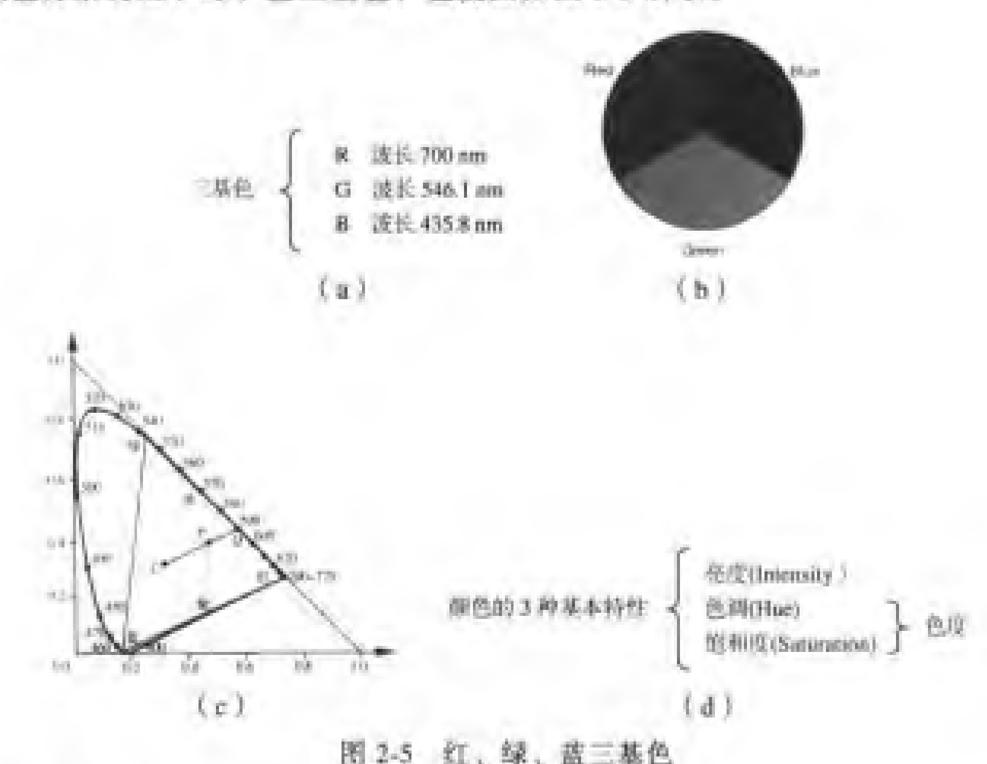


图2-5 红、绿、蓝三基色

2.1.4 色彩空间

一束普通白光，经过棱镜显示，可以分解为红、橙、黄、绿、青、蓝、紫。其中最基

本的是红（Red）、绿（Green）、蓝（Blue）由这三基色组成一个三维的色彩空间

1. 加法系统（RGB 三色原理）

三基色（RGB）的原理：自然界常见的各种颜色光，都是由红（R）（$\lambda=700\text{nm}$）、绿（G）（$\lambda=546\text{nm}$）、蓝（b）（$\lambda=435.8\text{nm}$）三种颜色光按照不同比例相配而成，同样，绝大多数颜色也可分解成红、绿、蓝三种单色光，这就是色度学中最基本的原理——三基色原理。

例如：

红色+绿色＝黄色

红色+蓝色＝品红

绿色+蓝色＝青色

红色+绿色+蓝色＝白色

当白光的亮度用 Y 来表示时，可用如下方程描述：

$$Y=0.299R+0.587G+0.114B \tag{2.1.1}$$

2. 减法系统（CMYK 四色系统）

充满颜色时，为黑色；没有颜色时，为白色。

使用 RGB 的补色做基本色：青（Cyan）、紫（Magenta）和黄（Yellow）。

C（青）＝W（白）－R（红）

M（紫）＝W（白）－G（绿）

Y（黄）＝W（白）－B（蓝）

CMYK 是印刷业的标准，在印刷时，用这种方法显示黑色，油墨很少能将颜色都吸收掉，深色效果较差，故加入一种黑色。

CMYK 空间比 RGB 空间小，所以彩色打印效果远不如彩色显示。

3. HSI 模式（HSV，HSB 模式）

艺术家使用的模式：

H（Hue）色调、S（Saturation）饱和度、B（Brightness）亮度

色调指光的颜色，如红、橙、黄、绿、青、蓝、紫为基色调。它是以单一波长得到的成分。

饱和度指色彩纯度的程度，加入的白光越多饱和度就越低。

亮度指彩色光对人眼引起的光刺激强度，它与光的能量有关。

HSB 是人类容易理解的模式，但由于设备限制，显示时要转为 RGB，打印时要转为 CMYK。

4. LAB 模式

LAB 模式是国际照明委员会（CIE）指定的颜色标准之一。

与 HSI 模式类似，它所用的参数是光强和色度。L 为光强度，A 保存从绿色到紫色所对应的色彩信息；而 B 保存从蓝色到黄色所对应的色彩信息。单个 A 或 B 无意义，只有 A，B 结合才有意义。

$$S=\sqrt{a^2+b^2} \tag{2.1.2}$$

$$H=\arctan\left(\frac{b}{a}\right) \tag{2.1.3}$$

LAB模式所的颜色空间最大，其他几个颜色空间都是它的子集。

5.　YIQ模式

美国国家制式委员会提出的NTSC彩色传输制式，使用YIQ三刺激值，它与RGB的关系是：

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & 0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1.4}$$

其中，Y为亮度，I，Q为色度信息。这种制式的好处是Y，I，Q信号的相关性小，便于传输。

颜色模型变换可用MATLAB函数实现。

2.2　数字图像文件的存储格式

数字图像在计算机中是以文件的形式存储的。常见的图像数据格式包括：BMP格式、TIFF格式、TGA格式、GIF格式、PCX格式以及JPEG格式等。

2.2.1　BMP格式的图像文件

BMP是Bitmap的缩写，意为“位图”。BMP格式的图像文件是微软公司特为Windows环境应用图像而设计的。BMP格式的主要特点有：

（1）BMP格式的图像文件以“.bmp”作为文件扩展名。

（2）根据需要，使用者可选择图像数据是否采用压缩形式存放。一般情况下，BMP格式的图像是非压缩格式。

（3）当使用者决定采用压缩格式存放BMP格式的图像时，使用RLE4压缩方式，可得到16色模式的图像；若采用RLE8压缩方式，则得到256色的图像。

（4）可以多种彩色模式保存图像，如16色、256色、24bit真彩色，最新版本的BMP格式允许32bit真彩色。

（5）数据排列顺序与其他格式的图像文件不同，从图像左下角为起点存储图像，而不是以图像的左上角作为起点。

（6）调色板数据结构中，RGB三基色数据的排列顺序恰好与其他格式文件的顺序相反。

BMP格式的图像文件结构可以分为文件头、调色板数据以及图像数据三部分，如图2-6所示。

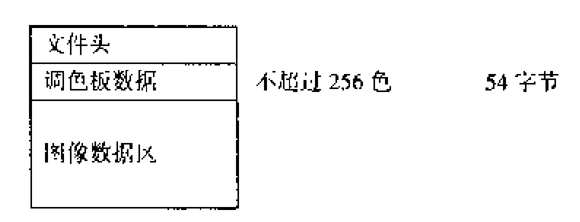


图2-6　BMP格式的图像文件结构

图像文件大小：

灰度图像文件大小 ≈ 文件头 + 像素个数 × 灰度级数

彩色图像文件大小 ≈ 文件头 + 像素个数 × 颜色数
颜色数（用于表示颜色的位数）:

16 色（24）色	4bits
256 色（28）色	8bits = 1byte
65536（216）色	16 bits = 2bytes
1677 万（224）色	24 bits = 3bytes

2.2.2 TIFF 格式的图像文件

TIFF 是 Tag Image File Format 的缩写，它由 Aldus 公司 1986 年推出，后来与微软公司合作，进一步发展了 TIFF 格式。

TIFF 格式的图像文件具有如下特点：

（1）TIFF 格式图像文件的扩展名是“.tif”。

（2）支持从单色模式到 32bit 真彩色模式的所有图像。

（3）不针对某一个特定的操作平台，可用于多种操作平台和应用软件。

（4）适用于多种机型，在 PC 计算机和 Macintosh 计算机之间，可互相转换和移植 TIFF 图像文件。

（5）数据结构是可变的，文件具有可改写性，使用者可向文件中写入相关信息。

（6）具有多种数据压缩存储方式，使解压缩过程变得复杂化。

TIFF 格式的图像文件结构如图 2-7 所示。

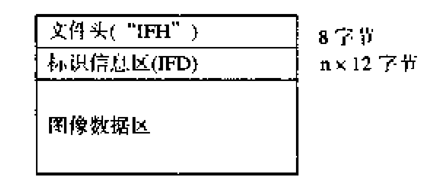


图 2-7 TIFF 格式图像文件的数据结构

文件头由 8 个字节组成。该文件头必须位于和 0 相对的位置，并且位置不能移动。在标识信息区（IFD）目录中，有很多由 12 个字节组成的标识信息，标识的内容包括指示标识信息的代号、数据类型说明、数据值、文件数据量等。图像数据区是真正存放图像数据的部分，该区的数据指明了图像使用何种压缩方法、如何排列数据、如何分割数据等内容。

2.2.3 GIF 格式的图像文件

GIF 是 Graphics Interchange Format 的缩写，它是 CompuServe 公司于 1987 年推出的。主要是为了网络传输和 BBS 用户使用图像文件而设计的。

GIF 格式的图像文件具有如下特点：

（1）GIF 格式图像文件的扩展名是“.gif”。

（2）对于灰度图像表现最佳。

（3）具有 GIF87a 和 GIF89a 两个版本。GIF87a 版本是 1987 年推出的，一个文件存储一个图像；GIF89 版本是 1989 年推出的很有特色的版本，该版本允许一个文件存储多个图像，可实现动画功能。

（4）采用改进的 LZW 压缩算法处理图像数据。

（5）调色板数据有通用调色板和局部调色板之分，有不同的颜色取值。

（6）不支持 24bit 彩色模式，最多存储 256 色。

GIF 格式的图像文件结构如图 2-8 所示。

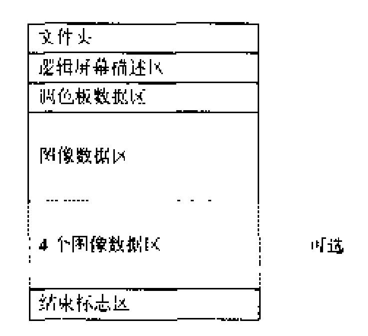


图 2-8 GIF 格式图像文件的数据结构

文件头是一个带有识别 GIF 格式数据流的数据块，用以区分早期版本和新版本。逻辑屏幕描述区定义了与图像数据相关的图像平面尺寸、彩色深度，并指明后面的调色板数据属于全局调色板还是局部调色板。若使用的是全局调色板，则生成一个 24bit 的 RGB 全局调色板，其中一个基色占用一个字节。

图像数据区的内容有两类，一类是纯粹的图像数据，一类是用于特殊目的的数据块（包含专用应用程序代码和不可打印的注释信息）。在 GIF89a 一格式的图像文件中，如果一个文件中包含多个图像，图像数据区将依次重复数据块序列。结束标志区的作用主要是标记整个数据流的结束。

2.2.4 JPEG 格式的图像文件

JPEG 是 Joint Photographic Experts Group 的缩写，该标准是国际标准化组织 ISO 的下属专家小组提出的。该格式文件采用有损编码方式，原始图像经过 JPEG 编码，使 JPEG 格式的图像文件与原始图像发生很大差别，但不易察觉。

JPEG 格式的图像文件具有如下特点：

（1）JPEG 格式图像文件的扩展名是“.jpg”。

（2）适用性广泛，大多数图像类型都可以进行 JPEG 编码。

（3）对于使用计算机绘制的具有明显边界的图形，JPEG 编码方式的处理效果不佳。

（4）对于数字化照片和表达自然景观的色彩丰富的图片，JPEG 编码方式具有非常好的处理效果。

（5）使用 JPEG 格式的图像文件时，需要解压缩过程。

JPEG 格式的图像文件一般有两种内部格式：

一种是目前被广泛使用的 JFIF 格式，它包含一个常驻的 JPEG 数据流，其作用是提供解码所需的数据，而不需要使用外部数据。另一种是 JPEG-in－TIFF 格式，该格式把 JPEG 图像压缩保存到 TIFF 格式的文件中，它在保存和读出时，很容易受外部条件的限制和影响，目前还未得到广泛的应用。

2.2.5 TGA 格式的图像文件

TGA 格式的图像文件由 Truevision 公司开发，最初的目的是支持本公司生产的 Targa

图形卡。该图形卡可以不借助调色板而直接显示 16M 种颜色（24 位真彩），是一流的计算机显示设备。

TGA 格式的图像文件目前的版本是 2.0 版。

TGA 格式的图像文件具有如下特点：

（1）TGA 格式图像文件的扩展名是“.tga”。

（2）支持任意尺寸的图像。

（3）支持 1 bit 单色到 32bit 真彩色模式的所有图像，具有很强的颜色表达能力，特别适合影视广播级的动画制作。

（4）图像的存储具有可选择性，图像数据既可以按照从上到下、从左到右的顺序进行存储，也可以相反的顺序存储。

（5）TGA 格式的图像对硬件的依赖性强，如果显示卡不具备 24bit 或 32bit 的显示能力，该格式的图像将不能正确显示。

TGA 格式的图像文件结构如图 2-9 所示。

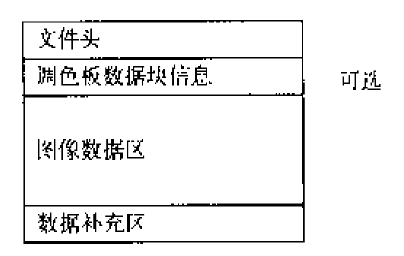


图 2-9　TGA 格式图像文件的数据结构

（1）文件头主要用于说明 TGA 文件的出处、颜色映像表类型、图像数据存储类型、图像数据存储顺序等内容。

（2）调色板数据块信息包括：调色板数据块构成方式、图像数据的组织方式等。

（3）图像数据区用于存储大量的图像数据，是描述图像的重要区域。

（4）数据补充区是 2.0 版本新增加的区域。其存储顺序为：开发者相关数据、开发者目录、扩充数据、数据块指针以及文件注脚。

2.2.6　PCX 格式的图像文件

PCX 由 PC Paintbrush 而得名，它由 Zsoft 公司推出，主要用于该公司开发的 PC Paintbrush 绘图软件。美国 Microsoft 微软公司后来将该绘图软件移植到 Windows 中，成为一个重要的功能模块。

PCX 格式的图像文件具有如下特点：

（1）PCX 格式图像文件的扩展名是“.pcx”。

（2）采用 RLE 压缩方式存储数据。

（3）拥有不同版本，分别用于处理不同显示模式下的数据。文件分为三类：单色文件、不超过 16 色的文件和 256 色文件。单色文件和 16 色的文件可不携带调色板数据，但 256 色文件则必须包含调色板数据。

（4）除了最新版本外，其他版本不支持 24bit 真彩色模式。

（5）图像显示与计算机硬件设备的显示模式有关。

PCX 格式的图像文件结构如图 2-10 所示。

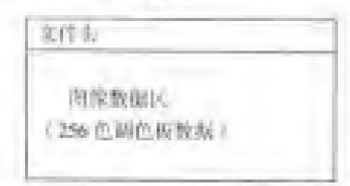


图 2-10　PCX 格式图像文件的数据结构

（1）文件头包含各种识别信息，其中包括 PCX 文件的特征信息、图像的大小和规模、调色板设置等。

（2）图像数据区用于表示图像，如果图像是 256 色模式，图像数据区的后面将存储 256 色调色板数据。

【例 2-4】同一幅图像各种存储格式的图像显示。

在常用的数字图像存储格式中，BMP 格式的图像是没有压缩的图像，所以图像信息没有丢失，TIFF 格式的图像有压缩形式（如图 2-11（b））和非压缩形式（如图 2-11（c）），GIF 格式的压缩效果最好，但是细节丢失也最多。下面通过图 2-11 对比同一幅图像的不同存储格式和图像的大小（存储量的大小）。

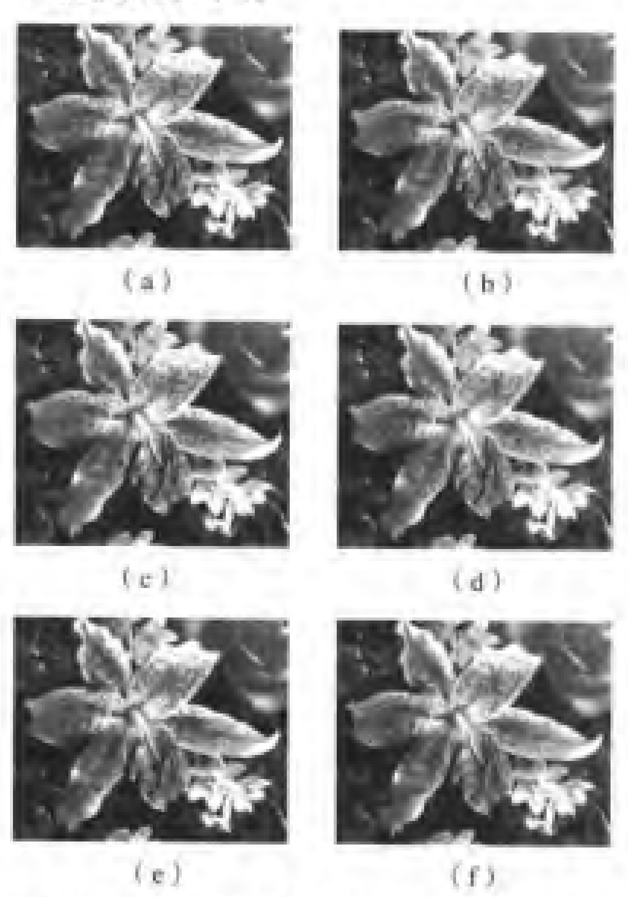


图 2-11　同一幅图像各种存储格式的图像显示

（a）BMP 图像压缩格式的图像。

（b）TIFF－NO－COMPRESS（无压缩）图像压缩格式的图像，大小为 125k。

（c）TIFF 图像压缩格式的图像，大小为 132k。

（d）JPG 图像压缩格式的图像，大小为 33.8k。

（e）PCX 图像压缩格式的图像，大小为 144k。

（f）GIF 图像压缩格式的图像，大小为 35.6k。

2.3 数字图像表示

我们获取图像的方法有很多种，其目标是相同的，就是要从感知的数据中产生数字图像。

为了产生一幅数字图像，需要把连续的感知数据转化为数字形式。在转化中包括两种处理：采样和量化。人们可以把数字图像看成离散采样点的集合（实际上也是如此），每个点具有各自的属性。这样对数字图像的处理就是对这些离散单元的操作。而取样和量化的结果是一个实际的矩阵。在本书中主要用两种方法来表示数字图像。当一幅图像 $f(x,y)$ 被采样，则产生的数字图像有 M 行和 N 列。现在坐标 (x,y) 的值变成离散量。为了表达清楚和方便起见，通常用整数表示离散的坐标。这样原点的坐标设为 $(x,y)=(0,0)$。沿图像的第一行的下一个坐标值用 $(x,y)=(0,1)$ 来表示，这样表示是很重要的，但并不表示这是图像的实际物理坐标值，只是一个贯穿本书的约定。现在以下面的紧凑的矩阵形式表示完整的 M ×N 数字图像：

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots\cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots\cdots & f(1,N-1) \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ f(M-1,0) & f(M-1,1) & \cdots\cdots & f(M-1,N-1) \end{bmatrix} \tag{2.3.1}$$

这个表达式的右侧定义了一幅数字图像。矩阵中的每个元素称为图像单元、图像元素或像素，一般简称为像素。

在某些讨论中，可以用传统的矩阵来表示数字图像和像素：

$$A=\begin{bmatrix} a_{0,0} & a_{0,1} & \cdots\cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots\cdots & a_{1,N-1} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ a_{M-1,0} & a_{M-1,1} & \cdots\cdots & a_{M-1,N-1} \end{bmatrix} \tag{2.3.2}$$

显然，$a_{i,j}=f(x=i,y=j)=f(i,j)$，所以式（2.3.1）和（2.3.2）是恒等矩阵。用更正规的数学术语描述取样和量化有时更有用。令 Z 和 R 分别表示整数集合和实数集合。采样的过程可以看作是把 xy 平面分为网格，每一网格的中心坐标是笛卡儿积 Z^2 的一对元素，Z^2 是所有有序元素对 (z_i,z_j) 的集合，z_j 是 Z 中的整数。因此，如果 (x,y) 是 Z^2 中的整数，并且 f 是把灰度级值（实数集合 R 中的一个实数）赋予每个特定坐标对 (x,y) 的函数，则 $f(x,y)$ 就表示一幅数字图像。这个函数赋值过程就是前面说的量化过程。如果灰度也是整数的（通常是这样的情况），则 Z 代替 R，数字图像变成一个二维函数，且它的坐标和幅值都是整数。数字化过程对于 M，N 值和每个像素允许的离散灰度级数 L 需要一个判定。对 M 和 N 除了必须取正数之外没有其他要求。然而出于处理、存储和采样的考虑，灰度级典型的取值是 2 的正数次幂：

$$L=2^k \tag{2.3.3}$$

这里假设离散灰度级是等间距的并且是区间 [0,L−1] 内的整数。有时灰度级取值范围也称为图像的动态范围。把占有灰度级全部有效段的图像叫做高动态范围图像。当相当大

数目的像素呈现这样的特性时，图像就有较高的对比度。相反，低动态图像看上去似乎是被冲淡了的灰暗格调。数字 b 是存储数字图像需要的比特数：

$$b = M \times N * k \quad (2.3.4)$$

当 M = N 时，上式变成：

$$b = N^2 k \quad (2.3.5)$$

2.4　空间和灰度级分辨率

采样值是决定一幅图像空间分辨率的主要参数。基本上，空间分辨率是图像中可辨别的最小细节。假定画一幅宽度为 2W 的垂直线的图，在线间还有宽度为 W 的线。线对是由一条与它紧邻的线组成。这样，线对的宽度为 2W，并且每单位距离由 1/2W 对线。广泛使用的分辨率的意义是在每个单位距离可以分辨的最小线数目，例如每毫米 100 线对。

类似地，灰度级分辨率是指在灰度级别中可分辨的最小变化。但是，在灰度级中，测量可分辨率的变化是一个高度主观的过程。这里考虑了用于产生数字图像采样数目的判断方法，但是，对于灰度级数，这种方法却不可行。由于硬件方面的考虑，灰度级数通常是 2 的整数次幂。大多数情况该值取 8 比特，在某些特殊的灰度增强的应用场合可能用到 16 比特。有时，我们寻求 10 或 12 比特精度数字化一幅图像的系统，但这些系统都是特例而不是常规系统。当没有必要对涉及像素的物理分辨率进行实际度量和在原始场景中分析细节等级时，通常就把大小为 M × N，灰度为 L 级的数字图像称为空间分辨率为 M × N 像素，灰度级分辨率为 L 的数字图像。为清晰起见，在以后的讨论中将经常使用这一术语，仅在需要时参照实际分辨率细节。

【例 2-5】在数字图像中各种采样数目的典型效果。

图 2-12 显示了一幅 512 × 512 像素的图像，其灰度用 8 比特表示。如图 2-12 所示的其他图像是对 512 × 512 图像采样的结果。采样是从原图像中删去一样的行和列数完成的。例如，256 × 256 图像是从 512 × 512 图像中每隔一行或一列删去一行或一列得到的，128 × 128 图像是从 256 × 256 图像中每隔一行或一列删去一行或一列得到的等等。其灰度级保持在 256。







512　256　128　64　32

图 2-12　采样数目典型效果图

这些图像显示了不同取样密度间大小比例，但是他们大小的差别是的很难看出减少去取样数目带来的影响。比较这一效果的最简单办法是，通过复制行或列使采样后的图像复原到 512×512 大小。其结果示于图 2-13 中。图 2-13 是将图 2-12 中采样后的图像进行复原得到的图像。事实上，将图 2-13（b）中的 256×256 图像和图 2-12 中 512×512 图像进行比较，很难看出两幅图像的区别。细节损失程度太细微以至于不可能看出这些图像在这一尺度上印刷出来后的区别。图 2-13（c）中的 128×128 图像稍微明显的颗粒也开始出现。这些效果在图 2-13（d）中的 64×64 图像和图 2-13（e）中的 32×32 图像已经非常明显了。

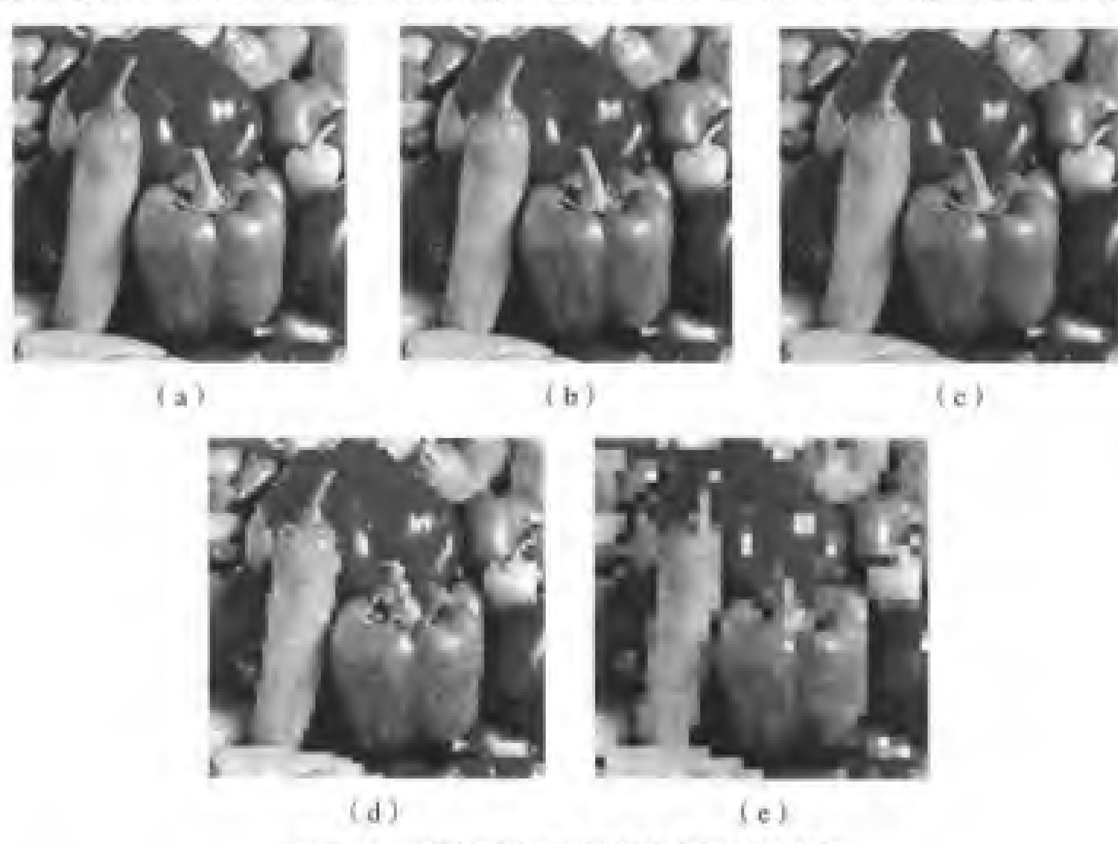


图 2-13 采样后的图像进行复原得到的图像

（a）512×512，8 比特图像。

（b）用复制行或者列的方法把 256×256 的图像复原为 512×512 的图像。

（c）用复制行或者列的方法把 128×128 的图像复原为 512×512 的图像。

（d）用复制行或者列的方法把 64×64 的图像复原为 512×512 的图像。

（e）用复制行或者列的方法把 32×32 的图像复原为 512×512 的图像。

【例 2-6】 在数字图像灰度值变化的典型效果。

在这个例子中，保持采样数恒定而以 2 的整数次幂方式把灰度级从 256 减少到 2。如图 2-14（a）所示是一幅大小为 640×480 的灰度图像，它以 k = 8（256 灰度级）显示。图 2-14（b）到（h）是在保持空间分辨率 640×480 不变的情况下，比特数从 k = 7 减少到 k = 1 而得到的图像。256、128、64 灰度级图像的全部实际效果都是在视觉上可分辨的。然而，如图 2-14（d）所示的 32 灰度级图像在平滑灰度区域内（特别在天空部分）有一组几乎看不见的非常细小的山脊状结构，这种效果是由在数字图像平滑区灰度级数量不足引起的，它被称之为伪轮廓，之所以这样叫是因为山脊像地图中的等值线。伪轮廓通常在平均

的平面上以 16 级和更少的灰度显示的图像中十分明显，如图 2-14（c）到（h）所示图像。

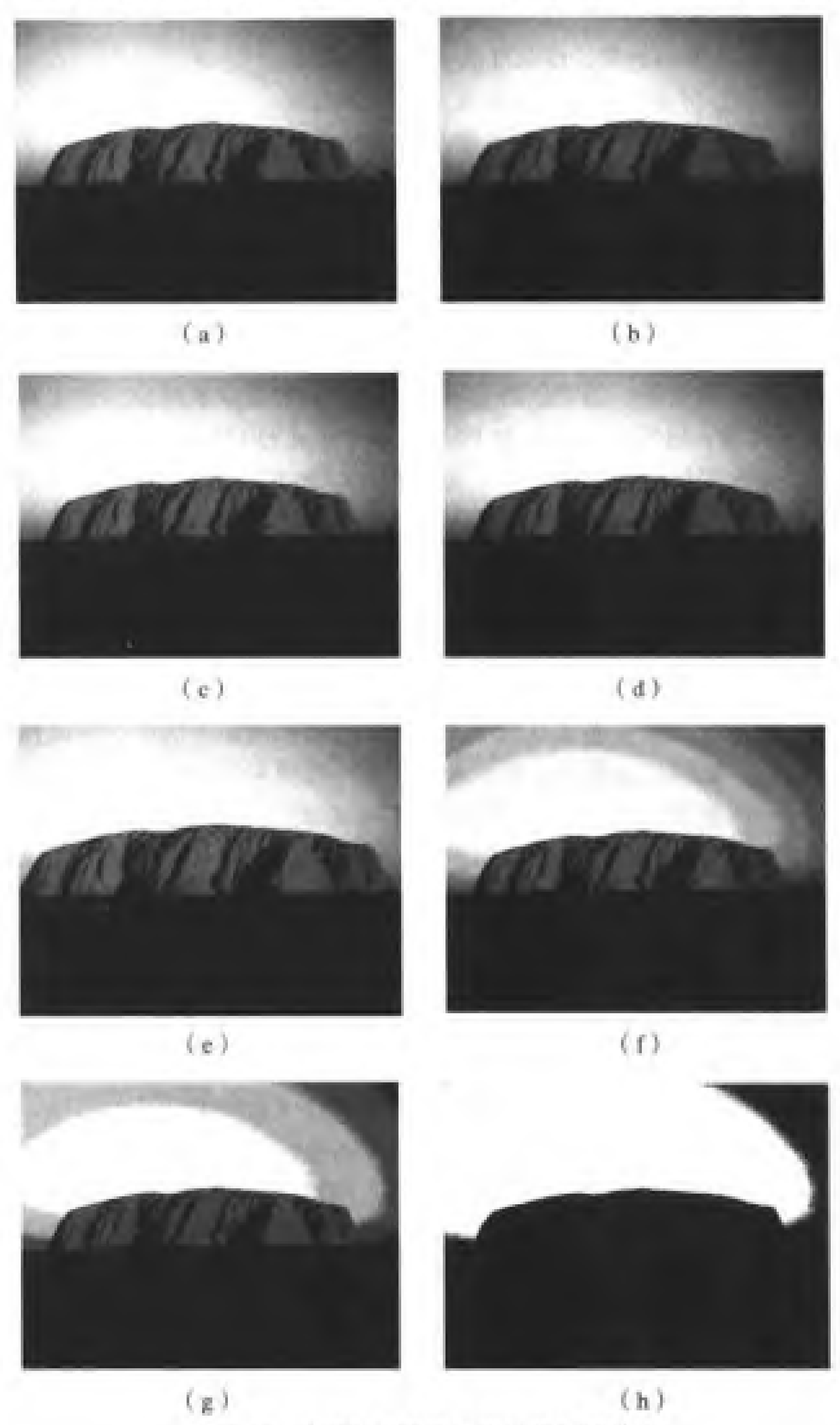


图 2-14　数字图像灰度值变化的典型效果

（a）640×480，256 灰度级图像。

（b）保持空间分辨率不变，以灰度级 128 显示的图像。

（c）保持空间分辨率不变，以灰度级 64 显示的图像。

（d）保持空间分辨率不变，以灰度级 32 显示的图像。
（e）保持空间分辨率不变，以灰度级 16 显示的图像。
（f）保持空间分辨率不变，以灰度级 8 显示的图像。
（g）保持空间分辨率不变，以灰度级 4 显示的图像。
（h）保持空间分辨率不变，以灰度级 2 显示的图像。

2.5 放大和收缩数字图像

用如何放大和收缩数字图像的简单讨论来阐述采样和量化。这个命题涉及到采样和量化，因为放大可以看作过采样，收缩可以看作欠采样。这两种操作与采样和量化一幅原始连续图像之间的关键区别是放大和收缩使用于数字图像。

放大要求执行两步操作：创立新的像素和对这些新位置赋灰度值。假定有一幅大小为 500×500 像素的图像，想要把它扩大为 1.5 倍，即 750×750 像素。概念上看，一种最容易的形象化放大法是在原始图像上放一个虚构的 750×750 的栅格。很显然，栅格间隔应该小于 1 格像素，因为是在一个较小的图像上去拟合它。为了对覆盖层上的任何点进行灰度赋值，需要在原图像上寻找最靠近的像素并把它扩展到原来指定的大小，以得到放大的图像，这种灰度赋值法叫做最近邻域插值（像素相邻在下一节讨论）。

产生图 2-15（b）到（f）图像所用的像素插值法是最近邻域插值法的一个特殊情况。当想要以整数倍增加图像的大小时，可采用像素复制的方法。例如，把图像增大一倍，可以复制每一行，这样，图像在垂直方向增大一倍。使用相同的方法可以任何整数倍增大图像（4 倍、5 倍等）。复制仅仅是为了达到希望的大小以要求的倍数去操作。每个像素的灰度赋值过程由这样的事实决定，即新位置的值准确复制旧位置的值。虽然最近邻域插值法较快，但它有不希望的特性，它会产生棋盘格效应，在高放大倍数时这一效应非常明显。图 2-15（e）到（f）是这一问题的典型例子。一种稍有改进的灰度赋值方法是采用四个最邻近点的双线性插值。令 (x', y') 代表放大图像中一点的坐标（把它想象为前边讨论的栅格上的点），并令 $v(x', y')$ 它被赋予的值。

对双线性插值来说，所赋灰度值由下式给出：

$$v(x', y') = ax' + by' + cx'y' + d \tag{2.5.1}$$

这里，四个系数由点 (x', y') 的四个最近邻点的四个未知方程决定。

图像收缩可以与刚才讨论过的放大相类似的方法操作。像素复制的等效过程是行－列删除。例如，把图像缩小，可以每隔一行（或一列）删除一行（或一列）。可以用放大栅格模拟非整数参数收缩，设想用扩大的栅格在原图像上拟合，做灰度级最近邻域或双线性插值，然后把栅格收缩到原来规定的大小。还可以使用更多邻点的插值。使用更多邻点意味着在更复杂表面对点赋灰度值，该方法通常会产生平滑一些的效果。对于三维图形和医学图像处理，这是一个特别重要的产生图像的方法。但是，对于通常目的的数字图像放大和收缩，减少一些额外的计算负担是合理的，因此双线性插值通常是首选插值方法。

【例 2-7】用双线性插值法放大图像。

图 2-15 的上面一行显示了将一幅 328×438 的灰度图像缩小后，再放大到原图像大小的结果，使用的是最近邻域插值法。用双线性插值方法的结果如图 2-15 的第二行。总的看来改进是比较明显的，特别是对于缩小 4 倍和 8 倍以后复原的图像。在缩小 15 倍后复原的

图像如图 2-15（f）所示，虽然比较模糊，但是比起图 2-15（c）来是好很多了。

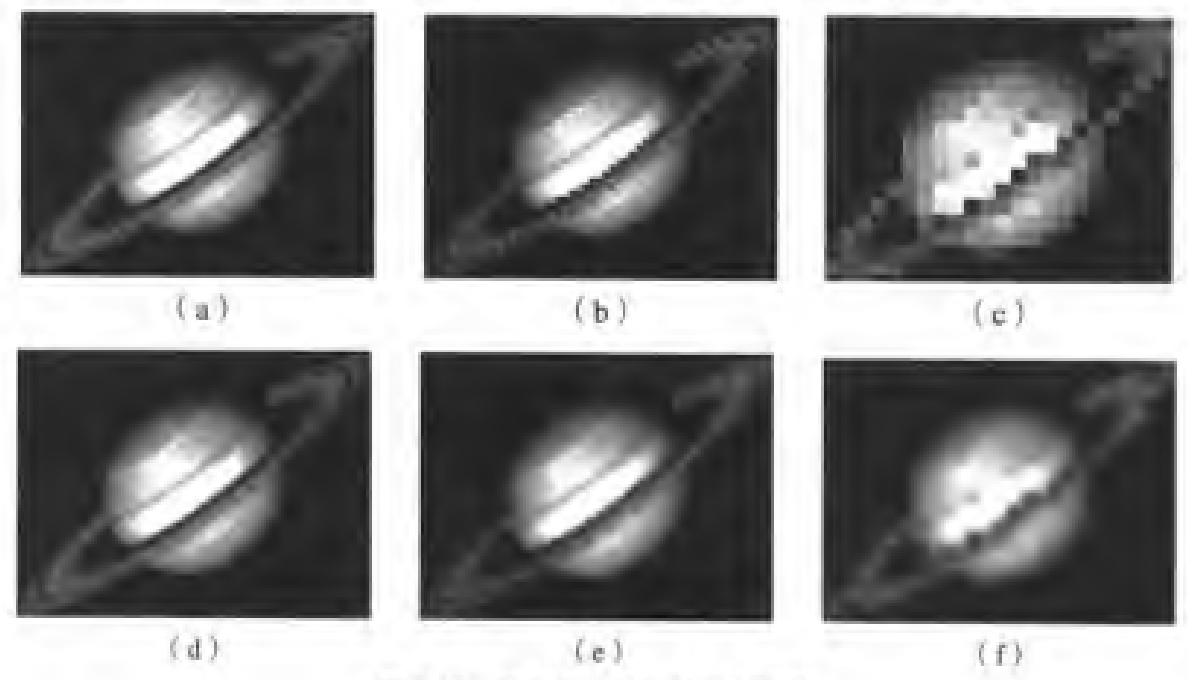


图 2-15 用双曲线插值法放大图像

（a）到（c）用最近邻域插值法将缩小 4 倍、8 倍和 16 倍的图像复原后的结果

（d）到（f）用双线性插值法将缩小 4 倍、8 倍和 16 倍的图像复原后的结果

2.6 像素间的基本关系

本节介绍一些数字图像中基本并且重要的像素之间的关系。用 $f(x, y)$ 表示一幅图像。为讨论方便，当指示 $f(x, y)$ 中某个特定的像素时，采用诸如 p 和 q 之类的小写字母表示。另外，$f(x, y)$ 中的像素子集用大写字母如 S、T 等来表示。

2.6.1 像素的邻域

对一个坐标为 (x, y) 的像素 p，它可以有四个水平和垂直的近邻像素，它们的坐标分别是 $(x+1, y)$、$(x-1, y)$、$(x, y+1)$、$(x, y-1)$。这些像素（用 r 表示）组成 p 的 4-邻域，记为 $N_4(p)$，如图 2-16（a）所示。坐标为 (x, y) 的像素 p 与它各个 4-邻域近邻像素是一个单位距离。如果 (x, y) 在图像的边缘，它的若干个近邻像素会落在图像外。

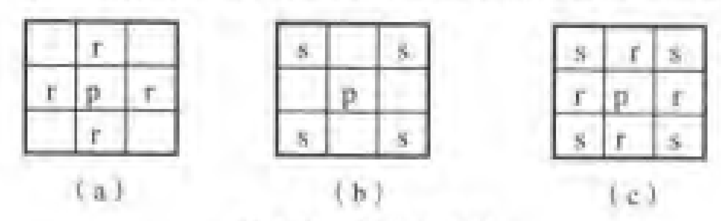


图 2-16 像素的邻域

像素 p 的四个对角近邻像素（用 s 表示）的坐标是 $(x+1, y+1)$、$(x+1, y-1)$、$(x-1, y+1)$、$(x-1, y-1)$。它们记为 $N_D(p)$，见图 2-16（b）。这些像素点再加上 p 的 4-邻域像素合称为 p 的 8-邻域，记为 $N_8(p)$，见图 2-16（c）。同上，如果 (x, y) 在图像的边

缘，$N_D(p)$ 和 $N_8(p)$ 中的若干个像素会落在图像外。

2.6.2 像素的连通性

像素间的连通性在建立图像中目标的边界和确定区域的元素时是一个重要的概念。它简化了许多数字图像概念和定义，如区域和边界。为了确定两个像素是否连通，必须确定它们是否相邻及它们的灰度值是否满足特定的相似性准则（或者说，它们的灰度值是否相等）。连通性可以进一步分成连接和连通，连接是连通的一种特例。要确定两个像素是否连接，需要在某种意义上确定它们是否接触（例如它们是否为 4-近邻像素）以及它们的灰度值是否满足某个特定的相似准则（例如它们灰度值相等）。举例来说，在一幅只有灰度 0 和 1 的二值图中，两个 4-近邻像素只有在它们具有相同的灰度值时才可以说是连接的。

下面用 V 表示定义连接的灰度值集合。例如在一幅二值图中，为考虑灰度值为 1 的像素之间的连通性，V = {1}。又知在一幅灰度图中，考虑具有灰度值在 8 到 16 之间像素的连通性，则 V = {8，9，…，15，16}。可定义以下三种连接情况（可参见图 2-16）：

（1）4-连接：两个像素 p 和 r 在 V 中取值且 r 在 $N_4(p)$ 中，则它们为 4-连接。

（2）8-连接：两个像素 p 和 r 在 V 中取值且 r 在 $N_8(p)$ 中，则它们为 8-连接。

（3）m-连接（混合连接）：2 个像素 p 和 r 在 V 中取值且满足下列条件之一，则它们为 m-连接：

① r 在 $N_4(p)$ 中。

② $N_4(p)\cap N_4(r)$ 是空集，这个集合是由 p 和 r 的在 V 中取值的 4-近邻像素组成的。

混合连接可认为是 8-连接的一种变型，引进它是为了消除使用 8-连接时常会出现的多路连接问题。图 2-17 给出一个示例。

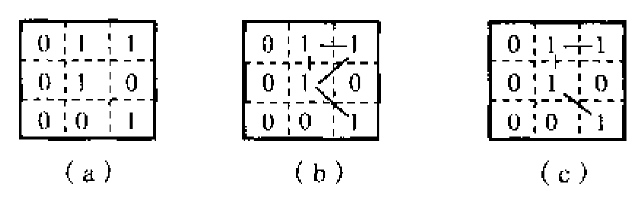


图 2-17 像素间的连接

考虑图 2-17（a）所示的像素排列，当 V = {1}时，中心像素的 8-近邻像素间的连接由图（b）中的连线所示。请注意由于允许 8-连接所产生的歧义性（即中心像素和右上角像素间有两条连线），这种歧义性当用 m-连接时就消除了（如图（c）所示），因为中心像素和右上角像素之间直接的 m-连接不能成立（第 1 和第 2 条件均不满足）。

如果一个像素 p 与另一个像素 q 相连接，则它们相毗邻。可根据所用的连接来定义 4-毗邻，8-毗邻或 m-毗邻。对两个图像子集 S 和 T 来说，如果 S 中的一些像素与 T 中的一些像素毗邻，则 S 和 T 是毗邻的。

从具有坐标(x,y)的像素 p 到具有坐标(s,t)的像素 q 的一条通路是由一系列具有坐标(x_0,y_0)，(x_1,y_1)，…，(x_n,y_n)的独立像素组成的。这里$(x_0,y_0)=(x,y)$，$(x_n,y_n)=(s,t)$，(x_i,y_i)与(x_{i-1},y_{i-1})毗邻，其中 1≤I≤n，n 为通路长度。如果$(x_0,y_0)=(x_n,y_n)$，则通路是闭合通路。可以依据特定的邻接类型定义 4，8 或者 m 邻接。

例如，如图 2-17（b）所示，东北角点之间的通路是八通路，图 2-17（c）中的通路是

m 通路。注意在 m 通路中不存在二义性。

设 p 和 q 是一个图像子集 S 中的两个像素，那么如果存在一条完全由在 S 中的像素组成的从 p 到 q 的通路，那么就称 p 在 S 中与 q 相连通。对 S 中任一个像素 p，所有与 p 相连通且又在 S 中的像素的集合（包括 p）合起来称为 S 中的一个连通分量。图像里相同连通分量中的两个像素互相连通，而不同连通组元中的各像素互不连通。如果 S 仅有一个连通分量，则集合 S 叫做连通集。

令 R 是图像中的像素子集。如果 R 是连通集，则称 R 为一个区域。一个区域 R 的边界（也称为边缘或轮廓）是区域中像素的集合，该区域有一个或多个不在 R 中的邻点。如果 R 是整幅图像（我们设该幅图像是像素的方形集合），则边界由图像的第一行、第一列和最后一行一列定义。这个附加定义是需要的，因为图像除了边缘没有邻点。正常情况下，当提到一个区域时，指的是一幅图像的子集，并且区域边界中的任何像素（与图像边缘吻合）都作为区域边界部分全部包含于其中。

边缘的概念在涉及区域和边界的讨论中常常遇到。然而，这些概念中由一个关键区别。一个有限区域的边界形成一条闭合通路，并且是“整体”概念。边缘是由具有某些导数值（超过预先设定的阈值）的像素形成。这样边缘的概念是基于在不连续点进行灰度级测量的局部概念。把边缘点连接成边缘线段是可能的，并且有时以与边界对应的方法连接线段，但并不总是这样。边缘和边界吻合的一个例外就是二值图像的情况。根据连通类型和所用的边缘算子，从二值区域提取边缘与提取区域边界是一样的，非常直观。在概念上，把边缘考虑为强度不连续的点和封闭通路的边界是有用的。

2.6.3　距离量度

给定三个像素 p，q，r，坐标分别为 (x,y)，(s,t)，(u,v)，如果下列条件满足的话，D 是距离量度函数：

（1）$D(p,q)\geqslant 0$　（$D(p,q)=0$ 当且仅当 $p=q$）。

（2）$D(p,q)=D(q,p)$。

（3）$D(p,r)\leqslant D(p,q)+D(q,r)$。

点 p 和 q 之间的欧氏（Euclidean）距离定义为：

$$D_E(p,q)=\left[(x-s)^2+(y-t)^2\right]^{\frac{1}{2}} \tag{2.6.1}$$

对于距离度量，距点 (x,y) 的距离小于或等于某一值 r 的像素是：中心在 (x,y) 半径为 r 的圆平面。点 p 和 q 之间的 D_4 距离，即城市（city-block）距离，定义为：

$$D_4(p,q)=|x-s|+|y-t| \tag{2.6.2}$$

在这种情况下，距 (x,y) 的 D_4 距离小于或等于某一值 r 的像素形成一个中心在 (x,y) 的棱形。例如，距 (x,y) 的 D_4 距离小于或者等于 2 的像素形成固定距离的下列轮廓：

$$\begin{matrix} & & 2 & & \\ & 2 & 1 & 2 & \\ 2 & 1 & 0 & 1 & 2 \\ & 2 & 1 & 2 & \\ & & 2 & & \end{matrix}$$

具有 $D_4=1$ 的像素是 (x,y) 的 4 邻域。

点 p 和 q 之间的 D_8 距离，即棋盘（Chessboard）距离，定义为：

$$D_8(p,q)=\max(|x-s|,|y-t|) \qquad (2.6.3)$$

在这种情况下，距 (x,y) 的 D_8 距离小于或等于某一值 r 的像素形成中心在 (x,y) 的方形。例如，距点 (x,y)（中心点）的 D_8 距离小于或等于 2 的像素形成下列固定距离的轮廓：

$$\begin{matrix} 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{matrix}$$

具有 $D_8=1$ 的像素是关于 (x,y) 的 8 邻域。

两点 p 和 q 之间的 D_4 距离等于它们之间最短的 4-通路的长度，D_8 距离也有类似性质。实际上考虑两点 p 和 q 之间的 D_4 距离和 D_8 距离时并不需要看它们之间是否真有一条通路，因为这些距离的定义只涉及这些点的坐标。但对 m-连通，两点之间的距离值（通路的长度）依赖于沿通路的像素和它们近邻像素的值。

【例 2-8】 m-连通时像素间距离测量示例。

在图 2-18 中，像素 p 和 q 的值均为 1，如果 s = 0，t = 0，则 $D_m(p,q)=2$；如果 s = 0，t = 1，则 $D_m(p,q)=3$；如果 s = 1，t = 0，则 $D_m(p,q)=3$；如果 s = 1，t = 1，则 $D_m(p,q)=4$。

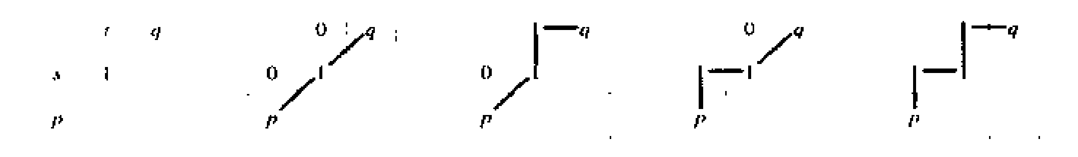


图 2-18　m-连通时像素间距离测量

2.7　算术和逻辑运算

在后续各章，大量介绍了关于像素间的运算，例如，用一幅图像去除另一幅图像。在式（2.3.2）中，图像以矩阵的形式表示，而矩阵除法无定义。因此当提到类似用一幅图像除另一幅图像的运算时，意思是在两幅图像相应的像素执行除法运算。例如，如果 f 和 g 是两幅图像，用 g 除 f 形成的图像的第一个像素值是用 g 中的第一个像素去除 f 中的第一个像素的结果，当然，假设在 g 中没有一个像素值为 0，其他的算术和逻辑操作也类似地定义为图像中相对应像素间的操作。

2.7.1　算术运算

（1）加法：平均法去噪声（见例 2-9）。

（2）减法：背景去除。

（3）乘（除）法：计算或去除非均匀光照。

【例 2-9】 平均法降噪。

算术运算一个非常好的应用就是图像平均降噪，一个静止场景的 M 幅图像，假定每幅

噪声图像来自于同一个互不相关的，噪声均值等于0的随机噪声图像的样本集，通过将M幅图像相加，然后求平均值。即对M幅图像进行平均，使图像中每一点的平方信噪比提高了M倍。M越大越好。这个推理将在第7章滤波器设计中做详细介绍。本例利用一幅静态的灰度图像，然后向灰度图像中加入椒盐噪声（椒盐噪声的随机性可以很好的模拟实验样本），将几十幅这样的加噪后的图像进行平均法去噪声，如图2-19所示，将10幅噪声图像均值降噪后如图2-19（c）所示，已经达到很不错的降噪效果，随着M值的增大，即样本图像越多，如图2-19（d）～（h）分别是20幅、30幅、40幅、50幅和100幅噪声图像均值降噪的结果，很明显可以看出，样本图像越多，效果越好。当样本图像达到50幅时，基本与原图没有什么差别，100幅时已经几乎达到了原图的清晰效果。当得不到清晰的原图时，通常用大量的静止场景的M幅图像进行平均降噪，以得到原图的近似图，这是一个非常有用的技术。

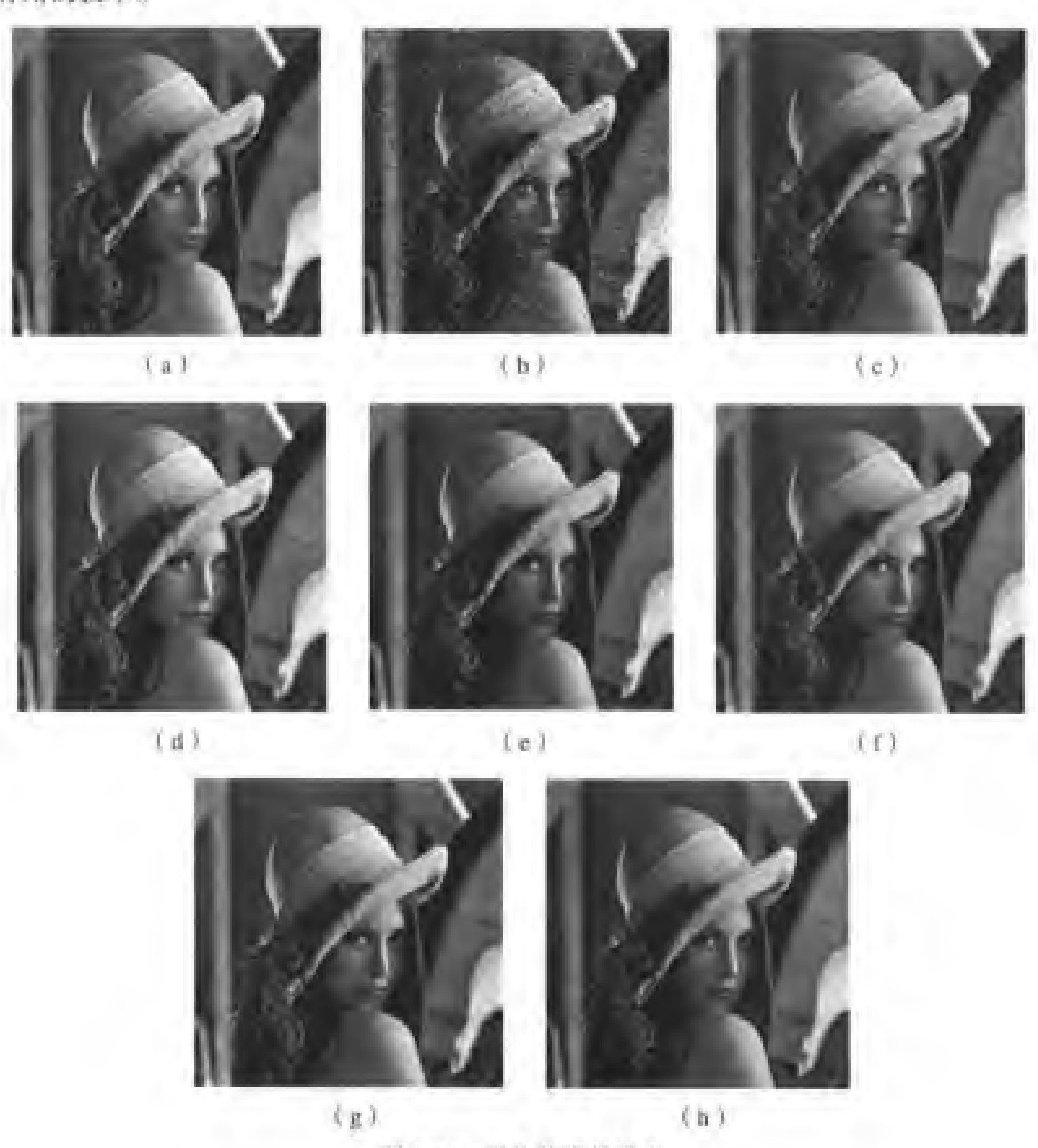


图2-19 平均值降低噪声

（a）原图（没有噪声）。

（b）加入“椒盐（salt & pepper）”噪声的图像。

（c）10 幅噪声图像均值降噪得到的图像。

（d）20 幅噪声图像均值降噪得到的图像。

（e）30 幅噪声图像均值降噪得到的图像。

（f）40 幅噪声图像均值降噪得到的图像。

（g）50 幅噪声图像均值降噪得到的图像。

（h）100 幅噪声图像均值降噪得到的图像。

本例子图片通过 MATLAB 图像处理软件处理得到，源程序如下：

```
A=imread('D:\MATLAB6p5\work\256x256\LENA256.BMP');
imshow(a)
[m,n]=size(a);
B=imnoise(A,'salt & pepper',0.03);
figure(2)
imshow(b)
B=double(b);
i=1;
while(i<20)
    C=imnoise(A,'speckle',0.03);
    C=double(c);
    for j=1:m
        for k-1:n
            B(j,k)=B(j,k)+C(j,k);
        end

    end
    i=i+1;
end

for x=1:m
    for y=1:n
        B(x,y)=B(x,y)/20;
    end

end

B=uint8(b);
figure(3)
imshow(b);
```

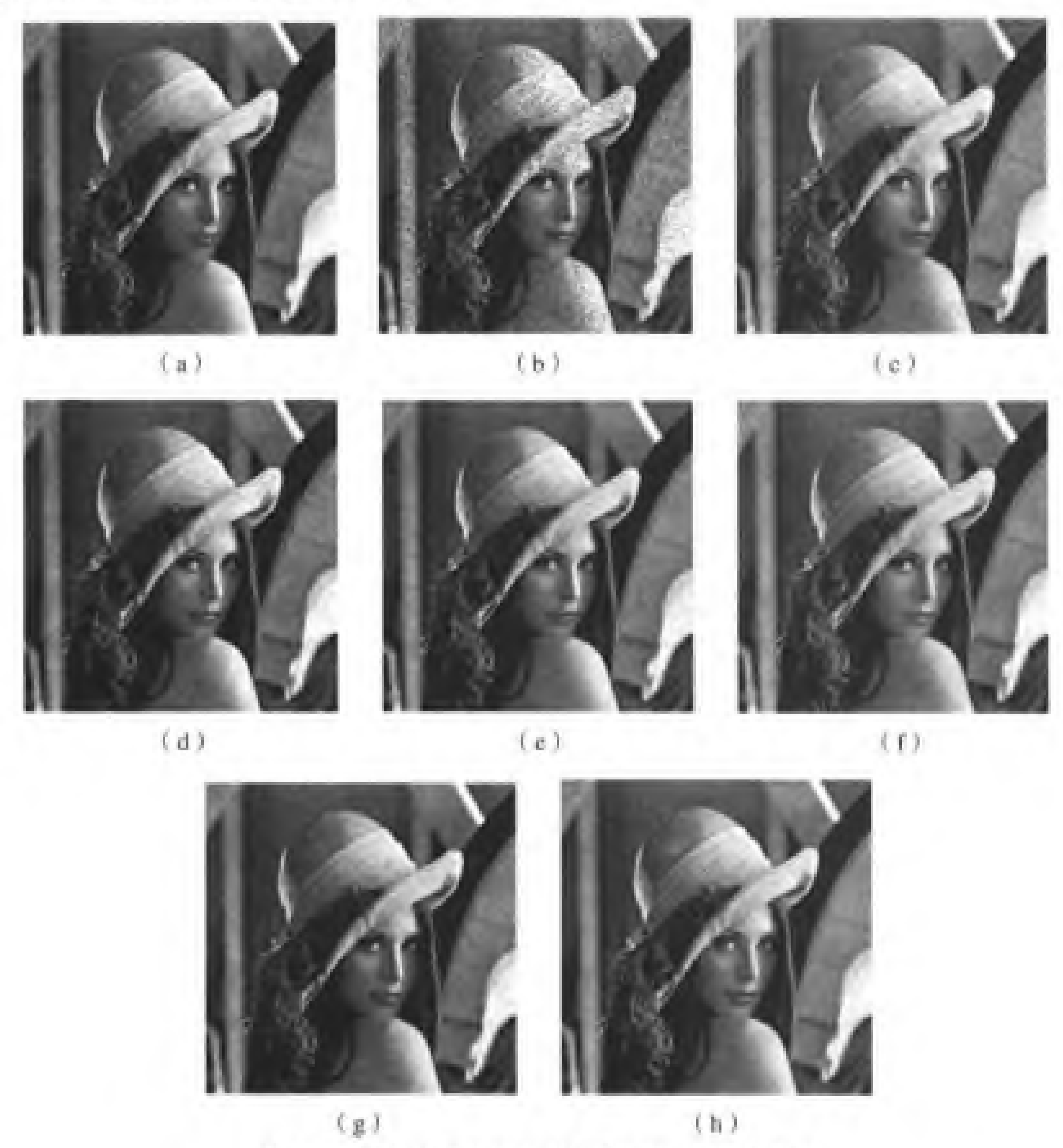
通过上例可以看到，对于加性噪声——“椒盐”噪声，均值降噪的效果是很好的，但对于乘性噪声又如何呢?

例如在图像中加入 speckle 噪声（均值为 0，方差为 V 的均匀分布随机噪声，参数 V 的默认值为 0.05），见例 2-10。

【例 2-10】平均降噪法应用于乘性噪声。

将上例中图像的噪声改为 speckle 噪声（乘性噪声），如图 2-20 所示，对于乘性噪声

speckle 噪声同样有着良好的去噪效果。



（a）　（b）　（c）

（d）　（e）　（f）

（g）　（h）

图 2-20　均值降噪法应用于被乘性噪声感染的图像

（a）原图（和例 2-9 一样）。

（b）加入 speckle 噪声（乘性噪声）的样本图像。

（c）~（h）分别是 10 幅到 100 幅样本图像均值降噪后的效果图。

2.7.2　逻辑运算

逻辑运算包括：

（1）AND（注意：不是像素本身的 AND）。

（2）OR。

（3）COMPLEMENT。

（4）XOR。

逻辑运算图解如图 2-21 所示。



图 2-21　逻辑运算图解

【例 2-11】加权平均法。

虽然将多幅图像平均求均值可以达到降噪的效果，但有时仅仅是简单的求平均值并不能达到很好的效果。另一种改进的方法就是加权平均，如下表方格中为像素点，方格中的值为权值，通过经验改变权值有时可以得到更好的效果。其计算公式如（2.7.1）式所示。

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$$z = w_1z_1 + w_2z_2 + \cdots + w_9z_9 = \sum_{i=1}^{9} w_iz_i \tag{2.7.1}$$

2.8　坐标变换

图像的平移、旋转和尺度变换是常见的图像坐标变换。在此，像素的值基本不变，仅几何位置发生变化。

图像平移：

$$\begin{aligned} X' &= X + X_0 \\ Y' &= Y + Y_0 \\ Z' &= Z + Z_0 \end{aligned} \tag{2.8.1}$$

矩阵形式：

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & X_0 \\ 0 & 1 & 0 & Y_0 \\ 0 & 0 & 1 & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.8.2}$$

归纳：

$$v' = Av \tag{2.8.3}$$

尺度（放缩）变换：

$$S = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8.4}$$

旋转变换：

坐标点绕 X 轴转 α 角度：

$$R_\alpha = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8.5}$$

坐标点绕 Y 轴转 β 角度：

$$R_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8.6}$$

坐标点绕 Z 轴转 γ 角度：

$$R_\gamma = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 & 0 \\ -\sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8.7}$$

变换矩阵级联：

$$v' = R_\gamma\left[S(Tv)\right] = Av \tag{2.8.8}$$

$$V' = AV \tag{2.8.9}$$

反变换：

$$T^{-1} = \begin{bmatrix} 1 & 0 & 0 & -X_0 \\ 0 & 1 & 0 & -Y_0 \\ 0 & 0 & 1 & -Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8.10}$$

$$R_\gamma^{-1} = \begin{bmatrix} \cos(-\gamma) & \sin(-\gamma) & 0 & 0 \\ -\sin(-\gamma) & \cos(-\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.8.11}$$

小结

本章的内容对后面的讨论提供了基本背景信息，对人类视觉系统的论述虽然简单，但是提供了人眼感知图像信息功能的一个基本概念。2.6 节介绍的像素间的基本关系是基于像素邻域处理技术的基本组成部分。

邻域处理方法是许多增强和复原技术的核心。在应用时，邻域处理由于处理速度和软硬件实现简单，很适合图像处理的商业应用。

最后，介绍了像素的算术和逻辑运算，以及坐标变换。其中逻辑运算在图像处理中是十分简单有用的，例如加法运算用于平均法降噪，对于类似“椒盐”噪声，去噪效果十分明显，同样对乘性噪声也有很好的效果。

重点： 了解数字图像处理技术的基础知识。

难点： 如何理解数字图像与一般图像的区别。

疑点： 图像处理技术有很多，如何根据具体的需要使用不同的处理技术。

练习二

一、选择题

1. 人眼对三种颜色最敏感，称为三基色，组成 RGB 三维彩色空间，这三种颜色是（　　）。

   A. 红、绿、蓝　　　　B. 红、黄、蓝

   C. 红、黄、绿　　　　D. 红、黄、紫

2. 光是电磁波的一种形式，频率为（　　）。

   A. 15~125Hz　　　　B. 5~125Hz

   C. 1050~102500Hz　　　　D. 105~1025Hz

二、思考题

1. 数字图像的文件存储格式有哪些？分别列出其主要特点。
2. 为什么研究数字图像处理技术需要心理学甚至生理学的知识？

三、上机题

编写一段程序，向一幅数字图像加入噪声。

第 3 章　线性系统

线性系统理论是一门成熟的理论学科，通常用于描述电路和光学系统的行为。它为采样、滤波以及空间分辨率的研究提供了坚实的数学基础。此外，线性系统理论对许多其他应用也很有帮助。

本章的主要内容有：

（1）引言。

（2）调谐信号与复信号。

（3）卷积。

（4）一些有用的函数。

（5）卷积滤波。

3.1　引言

本书把系统定义为接受一个输入并产生相应输出的任何实体。由于只关心输入和输出之间的关系，输入和输出可以是一维的、二维的或更高维的。在开始阶段将被限于两种情况：时间的一维函数和两个空间变量的二维函数。这样，以实际的物理过程为背景，可以使标记符号简单，分析不那么抽象，同时它还避免了由于过分追求通用而带来的麻烦。在必要时，这种分析可以推广到更高维。在本章的第一部分，讨论针对一维时间函数进行，并被推广到二维。

定义：

系统可以用泛函的方法表示：

（1）$y = f(x)$。其中 x、y 通常是一维或二维函数。

（2）$y(t) = F\{x(t)\}$。

（3）$x(t) \rightarrow y(t)$。

图 3-1 给出了一维的线性系统表示。在每种情况下，系统输入是一个或两个变量的函数，而系统产生的输出是相同变量的另一函数。

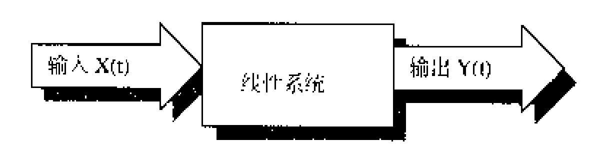


图 3-1　一维线性系统

图 3-2 给出了二维的线性系统表示。

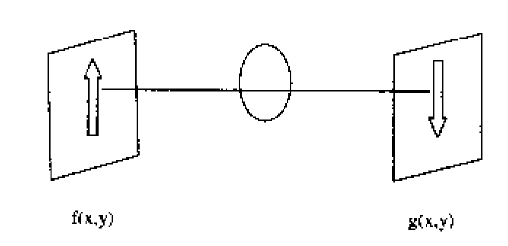


图 3-2　二维线性系统

线性（Linearity）系统的名字来源于如下的特征：设对某一个特定的系统，输入 $x_1(t)$ 产生输出 $y_1(t)$，即：

$$x_1(t) \to y_1(t) \tag{3.1.1}$$

（箭头表示“产生”），而另一端输入 $x_2(t)$ 产生输出 $y_2(t)$：

$$x_2(t) \to y_2(t) \tag{3.1.2}$$

则此系统是线性的，当且仅当它具有如下的性质：

$$x_1(t)+x_2(t) \to y_1(t)+y_2(t) \tag{3.1.3}$$

即先前两个信号的和作为输入产生的输出等于先前两个输出的和。任何不满足此约束条件的系统都是非线性的。非线性系统的分析在许多领域都有着广泛的应用，然而其复杂程度远远超过线性系统分析，就应用而言，并不要求这种附加的复杂性。因此讨论仅局限于线性系统的分析上面。

线性系统的定义说明，两个输入信号的和产生的输出等于这两个信号单独作用于系统所产生的输出的和。由此可以得到，若输入的信号乘以一个有理数，则输出信号将增加同样的倍数，即：

$$ax_1(t) \to ay_1(t) \tag{3.1.4}$$

无理数也同样成立，并以此作为一个公理。

式（3.1.1）、（3.1.2）、（3.1.3）和（3.1.4）定义了一个线性系统。应用泛函来定义在数学上更加严密，其定义如下：

线性：称系统具有线性，如果：

$y_1(t) = F\{x_1(t)\}$，　$y_2(t) = F\{x_2(t)\}$

且对任意常数 a_1,a_2，都有：

$a_1y_1(t) + a_2y_2(t) = F\{a_1x_1(t) + a_2x_2(t)\}$

这时采用线性系统理论对一个过程进行分析，其前提是此过程可以用（至少是近似）线性系统做模型。若被研究的系统不满足线性要求，那么它就是非线性的，则采用线性系统理论将得不到精确的结果，甚至反而有可能得到错误的结果。若系统仅有轻微的非线性，则它可被认为是线性系统而进行分析，但分析的结果仅在假设的范围内有效。

轻微非线性系统常采用线性系统理论来研究，这是因为线性系统理论易于处理和求解。然而，在对非线性系统进行如此处理时必须要小心，因为当假设条件不能得到满足时，线性系统理论也就失效了。在分析时，除了数学方法外，还要注意前提假设的有效性。

移不变性（shift invariance）其定义如下：

假设，对某线性系统有：

$$x(t) \to y(t) \tag{3.1.5}$$

现在让输入的信号沿时间轴平移 T，若满足下式：

$$x(t-T) \to y(t-T) \tag{3.1.6}$$

即输入信号除了平移了同样长度外，其他性质不变，则称系统具有移不变性。这样，对于移不变系统，平移输入信号仅仅使得输出信号移动同样的长度。重要的是输出的信号性质不变。

空间移不变性的二维推广：若输入图像相对于其原点有一平移，则输出图像除了相同的平移外其他不变。

3.2 调谐信号与复信号

在通常应用当中，可以将信号和图像表示为一个或两个变量的实值函数。函数值表示某物体参数的幅度，如随时间而改变的电压函数或随空间坐标改变的光的强度。若将输入和输出值推广到复（数）值函数，线性系统的性质将更加容易导出。由于实值函数可以看作复值函数的特例，所以这一推广并不会带来问题，相反它的好处将会在以后的讨论中显示出来。

3.2.1 调谐信号

考察如下形式的复值信号：

$$x(t)=e^{j\omega t}=\cos(\omega t)+j\sin(\omega t) \tag{3.2.1}$$

其中 $j^2=-1$，$x(t)$ 称为调谐信号。它是一个时间的复值函数，可以看作为一个在复平面内以角速度 ωt 旋转的单位向量，如图 3-3 所示。角频率 ω（单位为"弧度/秒"）与每秒的频率 f（单位为赫兹）满足关系 $\omega=2\pi f$ 。

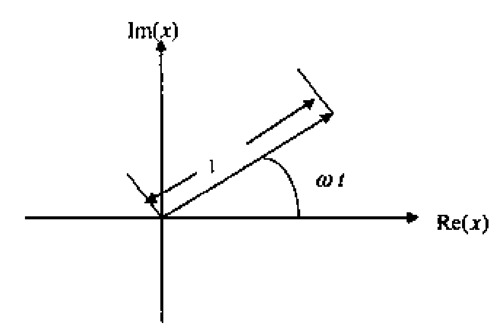


图 3-3 调谐信号的向量表示

3.2.2 对调谐输入的响应

设一个线性移不变系统，其输入信号为：

$$x_1(t)=e^{j\omega t} \tag{3.2.2}$$

系统的响应为：

$$y_1(t)=K(\omega,t)x_1(t)=K(\omega,t)e^{j\omega t} \tag{3.2.3}$$

其中：

$$K(\omega,t)=\frac{y_1(t)}{e^{j\omega t}} \tag{3.2.4}$$

是ω和t的复值函数，将其乘以 $e^{j\omega t}$ 时就得到 $y_1(t)$ 。所以，必有这样的 $K(\omega,t)$ 存在。接着将信号 $x_1(t)$ 平移，得到第二个信号，对于 $x_1(t)$ 的平移信号 $x_2(t)=x_1(t-T)$ ，有：

$$x_2(t)=e^{j\omega(t-T)}=e^{-j\omega T}e^{j\omega t}=e^{-j\omega T}x_1(t) \tag{3.2.5}$$

其中，$e^{-j\omega T}$ 为复常数。注意 $x_2(t)$ 仅是 $x_1(t)$ 和一个复常数的乘积。能得到这样的结果就是因为 $x_1(t)$ 是一个调谐信号。

线性系统对于 $x_2(t)$ 的响应为：

由线性可得：

$$y_2(t)=e^{-j\omega T}y_1(t)=K(\omega,t)e^{-j\omega T}x_1(t) \tag{3.2.6}$$

由移不变性可得：

$$\begin{aligned} y_2(t) &= y_1(t-T) \\ &= K(\omega,t-T)e^{j\omega(t-T)} \\ &= K(\omega,t-T)e^{-j\omega T}e^{j\omega t} \\ &= K(\omega,t-T)e^{-j\omega T}x_1(t) \end{aligned} \tag{3.2.7}$$

由公式（3.2.6）和（3.2.7）得：

$$K(\omega,t-T)e^{-j\omega T}x_1(t) = K(\omega,t)e^{-j\omega T}x_1(t) \tag{3.2.8}$$

于是，$K(\omega,t-T)=K(\omega,t)$ 对一切 T 成立，即 $K(\omega,t)$ 与 t 无关。

这时（3.2.3）式 $y_1(t)=K(\omega,t)x(t)$ 可改写为一般形式：

$$y(t) = K(\omega)x(t) \tag{3.2.9}$$

在式（3.2.4）中定义的一般二元函数 $K(\omega,t)$ 变成了一个只是依赖于频率变量 ω 的一元函数 $K(\omega)$。式（3.2.9）表明了如下重要性质，即线性移不变系统对于调谐信号的响应等于输入信号乘以一个依赖于频率的复数。注意一个调谐信号输入中产生相同频率的调谐信号的输出。

3.2.3 调谐信号与正弦性信号

当用线性系统来描述物理系统的行为时，输入和输出都用实值函数来表示是最方便的。这样，可对线性移不变系统再加一个约束，即它们保持实值性。这意味着实值输入只能产生实值的输出。

据此可知，这样的系统也保持虚值性，去掉一个复输入的虚部只是去掉相对应复输出的虚部。

记：

$$\begin{aligned} x(t) &= \mathrm{Re}\{x(t)\} + j\,\mathrm{Im}\{x(t)\} \\ y(t) &= \mathrm{Re}\{y(t)\} + j\,\mathrm{Im}\{y(t)\} \end{aligned} \tag{3.2.10}$$

那么：

$$x(t) \to y(t) \Rightarrow \begin{aligned} &\mathrm{Re}\{x(t)\} \to \mathrm{Re}\{y(t)\} \\ &\mathrm{Im}\{x(t)\} \to \mathrm{Im}\{y(t)\} \end{aligned} \tag{3.2.11}$$

这意味着调谐信号输入的实部和虚部相互独立地通过系统。

对线性系统保持实值性可以简化分析。例如，若输入的是一个余弦信号，则可以为其加上一个正弦虚部以构成一个调谐信号的输入，确定系统对此调谐输入的响应，去掉复值输出的虚部。这种间接方法能大大简化分析，因此被经常采用。

任何正弦型函数都可被认为是某个（惟一）调谐函数的虚部，这个方法可按如下方法计算线性系统对一个正弦型信号的响应：

（1）将输入的正弦型信号表示为一个调谐信号。

（2）计算线性系统对此调谐输入的响应。

（3）取输出的调谐输出的实部作为真正的输出。

上述求解的过程实际是一种变换求解法：即先将正弦型信号变换成调谐信号，在调谐输入的情况下解题，再把得到的调谐输出变换回正弦型信号。

此方法与利用对数计算相乘变换相类似：首先对乘数进行对数变换，将变换后的结果相加，再将得到的和值通过指数变换变回去得到所需的乘积。正像对数变换的情形一样，从正弦型信号到调谐信号的变换大大简化了线性系统的分析。

3.2.4 传递函数

由式（3.2.9）中定义的 $K(\omega)$ 函数叫做线性系统的传递函数，并足以完全描述系统的性质。对于线性移不变系统来说，传递函数包含了系统的全部信息。

将 $K(\omega)$ 写成极坐标的形式：

$$K(\omega)=A(\omega)e^{j\phi(\omega)} \tag{3.2.12}$$

其中 $A(\omega)$ 和 $\phi(\omega)$ 都为实数， $A(\omega)$ 为幅值， $\phi(\omega)$ 为幅角。

传递函数的作用是什么呢？

设输入为一余弦函数，令其为某调谐信号的实部：

$$x(t)=\cos(\omega t)=\mathrm{Re}\{e^{j\omega t}\} \tag{3.2.13}$$

系统对此调谐输入的响应为：

$$K(\omega)e^{j\omega t}=A(\omega)e^{j\phi}e^{j\omega t}=A(\omega)e^{j(\omega t+\phi)} \tag{3.2.14}$$

输出为：

$$\begin{aligned}y(t)&=\mathrm{Re}\{A(\omega)e^{j(\omega t+\phi)}\}=\mathrm{Re}\{A(\omega)[\cos(\omega t+\phi)+j\sin(\omega t+\phi)]\}\\&=A(\omega)\cos(\omega t+\phi)\end{aligned} \tag{3.2.15}$$

其中，$A(\omega)$ 为乘积增益因子，代表系统对输入信号的放大或衰减倍数。$\phi(\omega)$ 为相移角，其作用是将调谐输入信号的时间原点加以平移。

至此，得到线性移不变系统的三个重要性质：

（1）调谐输入总产生同频率的调谐输出。

（2）系统的传递函数——一个仅依赖于频率的复值函数，包含了系统的全部信息。

（3）传递函数对一调谐信号输入只产生两种影响——幅度的变化和相位的平移（时间原点的平移）。

3.3 卷积

3.3.1 卷积的由来

考察如图 3-1 所示的线性系统，若能得到说明输入 $x(t)$ 和输出 $y(t)$ 之间关系的一般表达式，则对于线性系统的分析大有帮助。现在考察另一个线性系统模型，即实函数形式线性移不变系统模型。记为：

$$y(t)=\int_{-\infty}^{\infty}f(t,\tau)x(\tau)d\tau \tag{3.3.1}$$

上述线性函数表达式（叠加积分）足够表达任何线性系统 $x(t)$ 和 $y(t)$ 之间的关系。对任何线性系统，必定能够找到一个二元函数使式（3.3.1）成立。但人们希望能够用一个一元函数来刻画线性系统。

为了简化，加入移不变性。假设系统是移不变的，则：

$$y(t-T)=\int_{-\infty}^{\infty}f(t,\tau)x(\tau-T)d\tau \qquad (3.3.2)$$

令 $t'=t-T$ 和 $\tau'=\tau-T$ ，则：

$$y(t)=\int_{-\infty}^{\infty}f(t+T,\tau+T)x(\tau)d\tau \qquad (3.3.3)$$

比较（3.3.1）与（3.3.3），得：

$$f(t,\tau)=f(t+T,\tau+T) \qquad (3.3.4)$$

对∀T 成立。即，只要 t 和 τ 的差不变，$f(t,\tau)$ 的值不变，故可令：

$$g(t-\tau)=f(t,\tau) \qquad (3.3.5)$$

从而式（3.3.1）变成：

$$y(t)=\int_{-\infty}^{\infty}g(t-\tau)x(\tau)d\tau \qquad (3.3.6)$$

这就是著名的卷积积分。至此可以知道：

（1）线性移不变系统的输出可通过输入信号与一表征系统特性的函数 g(t)的卷积得到。

（2）表征函数 g(t)叫作系统的冲激响应。

（3）系统保持实值性当且仅当 g(t)为一实值函数。

（4）有两种办法来表示一个线性移不变系统输入和输出之间的关系：

① 任何一个这样的系统都有一个复值的传递函数，其与调谐输入相乘就得到对应的调谐输出。

② 任何一个这样的系统都有一个实值的冲激响应，其与输入信号的卷积给出对应的输出。

一个线性系统如图 3-4 所示。

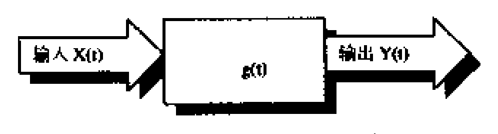


图 3-4 一个线性系统

线性移不变系统的传递函数和冲激响应有没有关系呢？既然移不变系统的传递函数和冲激响应都是惟一的，并且能够完全刻画系统，因此有理由推断，这两个函数之间存在着某种关系，这种关系将在以后给出。

3.3.2 一维卷积

将式（3.3.6）中的卷积积分简单记为：

$$y=g*x=\int_{-\infty}^{\infty}g(t-\tau)x(\tau)d\tau \qquad (3.3.7)$$

其中 * 用来表示两个函数的卷积。图 3-5 演示了卷积的运算过程。曲线 $y(t)$ 上的一点可以按照如下的方法获得：将函数 g 关于纵坐标反折，再向右平移距离 t，计算 x 和反折平

移后的 g 在各点的积，并将此积进行积分就得到了在 t 处的输出值。对每个 t 重复上述计算过程就可以得到输出曲线。当 t 变化时，反折的函数被平移通过静止不动的输入函数，$y(t)$ 的值取决于这两个函数的重叠部分的面积。

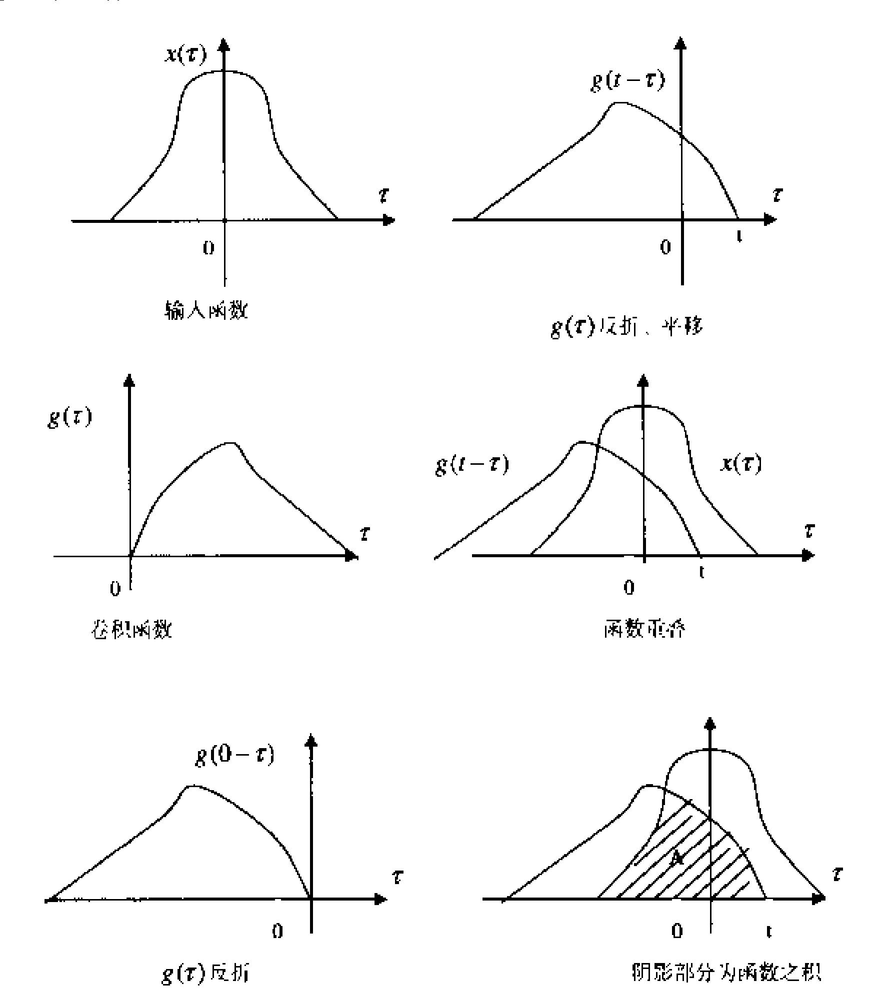


图 3-5 卷积运算过程

卷积运算有几个非常有用的性质：

（1）卷积具有交换性，即：

$$f * g = g * f \tag{3.3.8}$$

就是说将任一函数反折，都会得到相同的结果。

证明：对：

$$f * g = \int_{-\infty}^{\infty} f(\tau) g(t-\tau) d\tau \tag{3.3.9}$$

进行变量替换：

$$x = t - \tau \quad \tau = t - x \quad dx = -d\tau \tag{3.3.10}$$

经整理得到：

$$f * g = \int_{-\infty}^{\infty} f(t-x) g(x) dx = g * f \tag{3.3.11}$$

（2）卷积满足加法的分配律，即：

$$f*(g+h)=f*g+f*h \tag{3.3.12}$$

证明：

$$f*(g+h)=\int_{-\infty}^{\infty}f(t-\tau)[g(\tau)+h(\tau)]d\tau \tag{3.3.13}$$

整理得到：

$$f*(g+h)=\int_{-\infty}^{\infty}f(t-\tau)g(\tau)d\tau+\int_{-\infty}^{\infty}f(t-\tau)h(\tau)d\tau=f*g+f*h \tag{3.3.14}$$

（3）卷积满足结合律：

$$f*(g*h)=(f*g)*h \quad [=h*(f*g)] \tag{3.3.15}$$

这个性质由读者自己证明。

（4）卷积的求导法则：

$$\frac{d}{dt}[f*g]=f'*g=f*g' \tag{3.3.16}$$

3.3.3 离散一维卷积

对于离散序列，其卷积可用与连续函数相类似的方法求得。见式（3.3.17）。尽管离散卷积和连续卷积是相当不同的运算，但它们有很多共同的性质。尤其可以在数字图像上进行离散卷积运算，与连续卷积几乎具有对应的性质，而在许多图像数字化之前和变回连续形式之后对图像施加影响，都可以用连续卷积来描述。这一优点在图像恢复中被充分利用，图像恢复的目的是扭转已经加在图像上的退化影响。两个序列的卷积一般不能通过简单的向量乘法来实现。但若选择合适的表示形式，则卷积可用矩阵乘法来描述。

（1）定义。

设$\{f(i)\mid i=1,2,\ldots,m\}$和$\{g(j)\mid j=1,2,\ldots,n\}$分别是长度为 m 和 n 的序列，定义离散一维卷积是：

$$h(i)=f(i)*g(i)=\sum_{j=1}^{i}f(j)g(i-j+1) \tag{3.3.17}$$

其中，$h(i)$为长度是 N = m + n − 1 的序列。

$f(1) \quad f(2) \quad \ldots \quad f(m)$

$g(1) \quad g(2) \quad \ldots \quad g(n)$

f 和 g 的离散一维卷积为：

$$\begin{aligned}h(1)&=f(1)g(1)\\h(2)&=f(1)g(2)+f(2)g(1)\\h(3)&=f(1)g(3)+f(2)g(2)+f(3)g(1)\end{aligned} \tag{3.3.18}$$

……

① 卷积代表局部平均。

② 连续信号与离散信号的对应关系：

积分（连续信号）↔求和（离散信号）

自变量（连续信号）↔下标（离散信号）

③ 离散卷积保留了连续卷积的性质。

（2）矩阵形式。

① 将离散序列表示为向量可以充分利用成熟的线性代数运算。

② 将 f(i)拓展为周期为 N 的无限长周期序列，其中 N 为：N = m + n − 1。

③ 由于原 f(i)的长度为 m（m<N），因此不足部分需补 0，即：

$$f_p(i)=\begin{cases}f(i) & 1\leqslant i\leqslant m\\ 0 & m+1\leqslant i\leqslant N\end{cases} \tag{3.3.19}$$

同理：

$$g_p(i)=\begin{cases}g(j) & 1\leqslant j\leqslant n\\ 0 & n+1\leqslant j\leqslant N\end{cases} \tag{3.3.20}$$

这时：

$$h=G\bullet f=\begin{bmatrix}g_p(1) & g_p(N) & \cdots & g_p(2)\\ g_p(2) & g_p(1) & \cdots & g_p(3)\\ \cdots & \cdots & \cdots & \cdots\\ g_p(N) & g_p(N-1) & \cdots & g_p(1)\end{bmatrix}\begin{bmatrix}f_p(1)\\ f_p(2)\\ \cdots\\ f_p(N)\end{bmatrix} \tag{3.3.21}$$

其中 h 为 N 维列向量，矩阵 G 称为环矩阵，因其每一行都由上一行向右循环移位得到，G 中的每一行产生输出序列中的一个元素。这种结构可以将卷积写成矩阵的形式。

3.3.4　二维卷积

二元连续函数的卷积与一维情况相类似。不过要注意的是，在将讨论推广到二维时，将用 x 和 y 两个独立的变量。二维卷积的表达式如下：

$$h(x,y)=f*g=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(u,v)g(x-u,y-v)dudv \tag{3.3.22}$$

它可用图 3-6 表示。

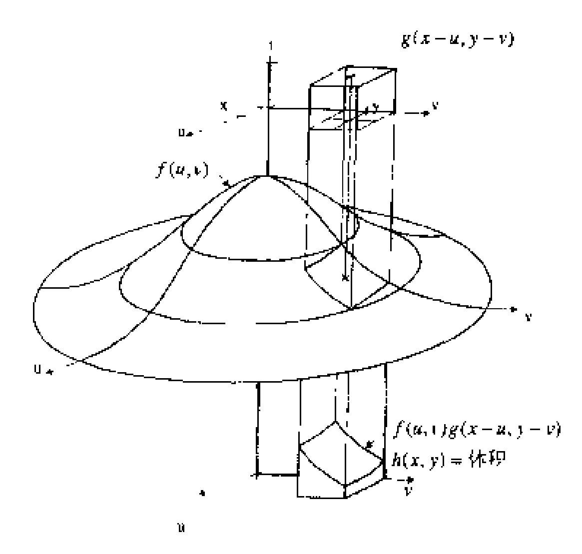


图 3-6　二维卷积

计算步骤：

（1）$g(0-u,0-v)$ 即：$g(u,v)$ 绕原点转 180 度。

（2）$g(x-u,y-v)$ 即：将旋转后的原点平移到 (x,y) 点。

（3）两个函数逐点相乘再求积分（即求交叠的体积）。

例如：令：

$$f(x,y)=Ae^{(x^2+y^2)/2a^2} \tag{3.3.23}$$

$$g(x,y)=\begin{cases}1 & -1\leqslant x\leqslant 1,-1\leqslant y\leqslant 1\\ 0 & otherwise\end{cases} \tag{3.3.24}$$

一个二维矩形脉冲与一个比它大的二维高斯函数做卷积。由于 $g(x,y)$ 关于原点对称，因此旋转 180 度后保持不变。$h(x,y)$ 的值就是当正方形脉冲至位置 (x,y) 时乘积函数的体积。

3.3.5 离散二维卷积

1. 定义

数字图像的卷积与连续函数的情形相类似，所不同的是其自变量取整数值，双重积分改为双重求和。这样对于图像 F，特征函数 G，离散二维卷积可写为：

$$H=F*G,\quad \text{即}\ H(i,j)=\sum_{m=1}^{i}\sum_{n=1}^{j}F(m,n)G(i-m+1,j-n+1) \tag{3.3.25}$$

由于 F 和 G 在有限范围内非零，所以求和计算只需要在非零部分重叠的区域上进行。G 叫做卷积核，与比它大的数字图像进行卷积。显然需要的乘法核加法的操作等于 G 中的像素与 F 中的像素数目之积（忽略图像边缘处的影响）。除非卷积核很小（或只有相对较小的非零区域），否则卷积运算将耗费相当的计算时间。

离散二维卷积对边界的处理方法：

（1）重复边缘值：通过重复图像边缘上的行和列，对输入图像进行扩充，使卷积在边缘可计算。

（2）卷绕输入图像：使之称为周期性，假设图像第一列紧接着最后一列。

（3）补零：在输入图像外部填充常数（例如补零）。

（4）忽略：去掉不能计算的行列，仅对可以计算的像素进行卷积。

第一种和第三种方法较为常用。在量化图像时最好使重要的信息不要落在距边缘小于卷积核宽一般的区域内，这样选用何种卷积都不会产生严重的后果。

2. 矩阵形式

（1）离散一维卷积。

将 f(i)和 g(j)分别周期化为 $f_p(i)$ 和 $g_p(i)$：

$$f_p(i)=\begin{cases}f(i) & 1\leqslant i\leqslant m\\ 0 & m+1\leqslant i\leqslant N\end{cases}$$

$$g_p(i)=\begin{cases}g(i) & 1\leqslant i\leqslant n\\ 0 & n+1\leqslant i\leqslant N\end{cases} \tag{3.3.26}$$

这时离散一维卷积的矩阵形式为：

$$h=G\bullet f=\begin{bmatrix}g_p(1) & g_p(N) & \cdots & g_p(2)\\ g_p(2) & g_p(1) & \cdots & g_p(3)\\ \cdots & \cdots & \cdots & \cdots\\ g_p(N) & g_p(N-1) & \cdots & g_p(1)\end{bmatrix}\begin{bmatrix}f_p(1)\\ f_p(2)\\ \cdots\\ f_p(N)\end{bmatrix} \tag{3.3.27}$$

其中，G 为环矩阵。

（2）离散二维卷积的矩阵形式。

正如一维离散序列，数字图像也可以方便地用矩阵来表示，这样可以利用线性代数的知识。同样矩阵运算不能直接用于执行卷积操作，但可以通过合适地构造来实现。

设数组 F 大小为 $m_1 \times n_1$，G 为 $m_2 \times n_2$，拓展 F 和 G，使其在 X 和 Y 方向分别是周期的，周期长度分别为 M 和 N，其中 $M \geqslant m_1 + m_2 - 1$，$N \geqslant n_1 + n_2 - 1$，可假设 M = N，不足之处补 0。扩展后新矩阵为 F_p 和 G_p。将 F_b 变成 $N^2 \times 1$ 维列向量 f_p。将 G_p 的第 i 行，用（3.3.27）的方法生成 $N \times N$ 循环矩阵 G_i，共有 N 个这样的矩阵 G_i（$1 \leqslant i \leqslant N$）。

$$G_b = \begin{bmatrix} [G_1] & [G_N] & \dots & [G_2] \\ [G_2] & [G_1] & \dots & [G_3] \\ \dots & \dots & \dots & \dots \\ [G_N] & [G_{N-1}] & \dots & [G_1] \end{bmatrix}_{N^2 \times N^2} \qquad (3.3.28)$$

G_b 为 $N \times N$ 的块矩阵。

这时，式（3.3.25）的二维卷积可以写成：

$$h_p = G_b f_p \qquad (3.3.29)$$

h_p 为 $N^2 \times 1$ 向量 $\Rightarrow N \times N$ 矩阵，输出图像。

【例 3-1】计算 F 和 G 的卷积。

其中：

$$F = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad G = \begin{bmatrix} -1 & 1 \\ -2 & 2 \end{bmatrix} \qquad (3.3.30)$$

解：N = 3

$$F_p = \begin{bmatrix} 1 & 2 & 0 \\ 3 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{N \times N}, \quad G_p = \begin{bmatrix} -1 & 1 & 0 \\ -2 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{N \times N} \qquad (3.3.31)$$

其中，$N = 2 + 2 - 1 = 3$

$$h_p = G_b f_p = \begin{bmatrix} -1 & 0 & 1 & 0 & 0 & 0 & -2 & 0 & 2 \\ 1 & -1 & 0 & 0 & 0 & 0 & 2 & -2 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 2 & -2 \\ -2 & 0 & 2 & -1 & 0 & 1 & 0 & 0 & 0 \\ 2 & -2 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 2 & -2 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 0 & 2 & -1 & 0 & 1 \\ 0 & 0 & 0 & 2 & -2 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 1 & -1 \end{bmatrix} \bullet \begin{bmatrix} 1 \\ 2 \\ 0 \\ 3 \\ 4 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ 2 \\ -5 \\ -3 \\ 8 \\ -6 \\ -2 \\ 8 \end{bmatrix} \qquad (3.3.32)$$

$$H = F * G = \begin{bmatrix} -1 & -1 & 2 \\ -5 & -3 & 8 \\ -6 & -2 & 8 \end{bmatrix} \qquad (3.3.33)$$

3.3.6 卷积的应用

数字线性滤波在以下三个主要的图像处理领域中十分有用：

（1）去卷积，即去除不需要的，但已经对图像施加了的线性系统影响。一个实例即利用卷积恢复由于透镜系统或运动所造成的模糊，这两种影响都可以被认为是由线性系统带来的。

（2）去除噪声，即去掉线性叠加在图像上的噪声信号。例如：

① 估计未受噪声污染前的信号。

② 检测噪声背景下是否存在已知特征。

③ 去除相干（周期）噪声。

（3）特征增强以削弱景物中的其他特征为代价来增强指定的特征（如边、点）的对比度。

3.4 一些有用函数

在线性系统理论和图像处理应用中，常用到五个函数。在本节中，下面将介绍这五个函数并给出他们的性质。对这些函数的掌握将有利于以后章节中内容和例子的理解。在本章的其余部分，继续使用 x 作为自变量。

3.4.1 矩形脉冲

定义矩形脉冲为：

$$\Pi(x)=\begin{cases}1, & -\frac{1}{2}<x<\frac{1}{2}\\ \frac{1}{2}, & x=\pm\frac{1}{2}\\ 0, & \text{其他}\end{cases} \tag{3.4.1}$$

$A\Pi\left(\frac{x}{a}\right)$ 表示高度为 A，宽度为 a 的矩形脉冲。如图 3-7 所示。该函数常作矩形采样窗和平滑函数。

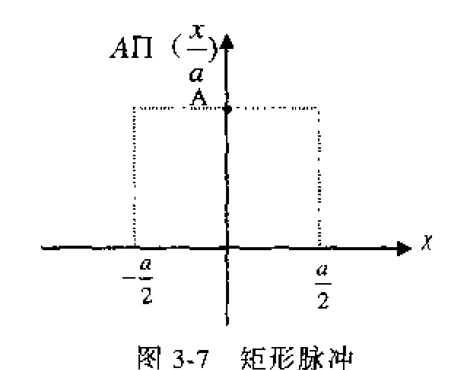


图 3-7 矩形脉冲

3.4.2 三角脉冲

定义三角脉冲：

$$\Lambda(x)=\begin{cases}1-|x|, & |x|\leqslant 1\\ 0, & |x|>1\end{cases} \tag{3.4.2}$$

例：$B\Lambda(\frac{x}{b})$ 表示高度为 B，宽度为 $2b$ 的三角脉冲，如图 3-8 所示。

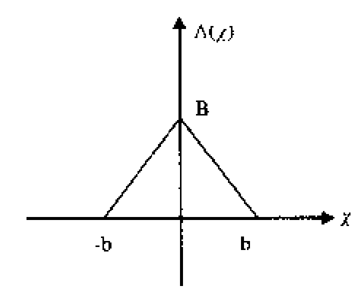


图 3-8 三角脉冲

两个相同矩形脉冲的卷积就得到一个三角脉冲：$\Lambda(x)=\Pi(x)*\Pi(x)$

3.4.3 高斯脉冲

（1）定义高斯脉冲为：

$$f(x)=e^{-x^2/2\sigma^2} \tag{3.4.3}$$

其形状如图 3-9 所示。

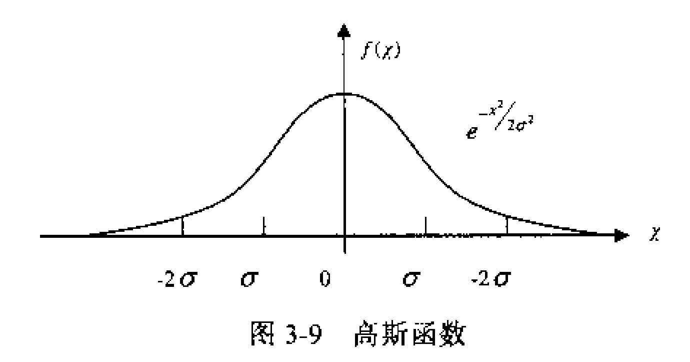


图 3-9 高斯函数

高斯函数的面积等于：

$$\int_{-\infty}^{\infty} e^{-x^2/2\sigma^2}dx=\sqrt{2\pi\sigma^2} \tag{3.4.4}$$

在概率论中，均值为 x_0 的标准正态分布为：

$$P(x)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-x_0)^2}{2\sigma^2}} \tag{3.4.5}$$

其中，x_0 为均值，σ^2 叫方差，σ 叫做标准差。即将高斯函数调整到单位面积。表 3-1 给出了高斯函数在某些点的值。

表 3-1 高斯函数值

x	$e^{-x^2/2\sigma^2}$
0	1
0.5σ	0.8825
1.0σ	0.6065
1.177σ	0.5000
1.5σ	0.3247
$1.177\sigma\sqrt{2}$	0.2500

续表 3-1

x	$e^{-x^2/2\sigma^2}$
2.0σ	0.1353
3.0σ	0.0111

（2）高斯函数的性质。

① 两个高斯函数的卷积是高斯函数。

$$Ae^{-(x-a)^2/2\sigma_1^2} * Be^{-(x-b)^2/2\sigma_2^2} = ABe^{-(x-c)^2/2\sigma_3^2} \tag{3.4.6}$$

其中：

$$c = a+b \text{ 和 } \sigma_3^2 = \sigma_1^2 + \sigma_2^2 \tag{3.4.7}$$

因此得到的高斯函数变宽了，其标准差是两个高斯函数方差之和的平方根，相对于原点的偏移是两原函数偏移之和。峰值的幅度是两原高斯函数幅度之积。

② 单峰形，常用于描述采样点、显示点。

③ 高斯函数有光滑（有无穷级导数），常用于图像的光滑化。

$$\frac{d}{dt}[f * g] = f' * g = f * g' \tag{3.4.8}$$

3.4.4 冲激函数

1. 定义

（1）单位冲激函数 $\delta(x)$（狄拉克 delta 函数），是广义函数，不是传统意义的函数。单位冲激函数的矩形脉冲模型如图 3-10 所示。平移冲激的模型如图 3-11 所示。

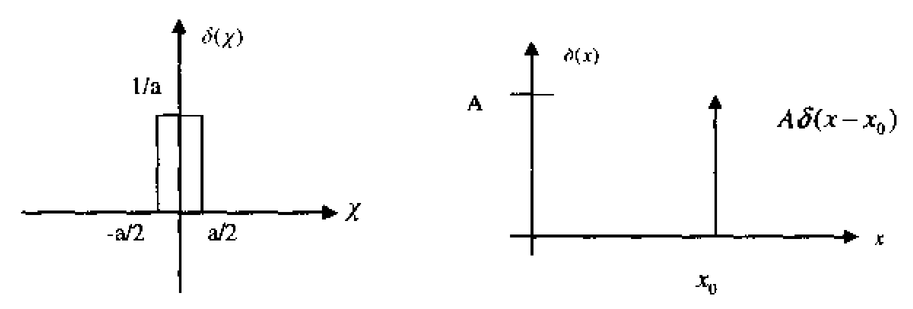


图 3-10 单位冲激函数的矩形脉冲模型　　图 3-11 平移冲激的图示

（2）它是通过积分性质来定义的积分函数：

$$\int_{-\infty}^{\infty} \delta(x)dx = \int_{-\varepsilon}^{\varepsilon} \delta(x)dx = 1 \tag{3.4.9}$$

其中，$\varepsilon > 0$ 的任意小的数。且当 $x \neq 0$ 时 $\delta(x) = 0$；在零点处单位冲激函数没有意义。由于 $\delta(x)$ 不是一个函数，其上述用法从某种意义上损害了数学理论的严密性。尽管可以用分布论对单位冲激函数做更进一步的更严密的探讨，但是这样引入更复杂的概念对应用并无太大的帮助。这里只是根据一般的工程时间来把 $\delta(x)$ 看作一个函数，注意其特殊性。

（3）单位冲激函数的描述：

$$\delta(x) = \lim_{a \to 0} \frac{1}{a} \Pi(\frac{x}{a}) \tag{3.4.10}$$

① 当 a 越小，脉冲越窄，越高，保证单位面积不变。

② 在极限情况下，脉冲变得无限高和无限窄。

由（3.4.9）式，有：

$$\int_{-\infty}^{+\infty} A\delta(x)dx = A \qquad (3.4.11)$$

进一步有， $\int_{-\infty}^{\infty} f(x)\delta(x)dx = \int_{-\varepsilon}^{\varepsilon} f(x)\delta(x)dx = \int_{-\varepsilon}^{\varepsilon} f(0)\delta(x)dx$

$$= \int_{-\infty}^{\infty} f(0)\delta(x)ds = f(0) \qquad (3.4.12)$$

这是因为单位冲激函数在非零处取零值。这个更一般的积分性质常用作单位冲激函数的定义。

2. 冲激函数的性质

单位冲激函数具有筛选性，它可以孤立曲线上的单个点。由下式可以看出：

$$\int_{-\infty}^{\infty} f(x)\delta(x-x_0)dx = \int_{-\infty}^{\infty} f(x+x_0)\delta(x)dx = f(x_0) \qquad (3.4.13)$$

当让一个函数与平移后的单位冲激函数相乘并对结果进行积分后，得到的应该是该函数在单位冲激处的函数值。实际上作变量替换 $\tau = x - x_0$，有 $dx = d\tau$，带入式（3.4.13）有：

$$\int_{-\infty}^{\infty} f(x)\delta(x-x_0)dx = \int_{-\infty}^{\infty} f(\tau+x_0)\delta(\tau)dx$$

而根据式（3.4.12），有：

$$\int_{-\infty}^{\infty} f(x+x_0)\delta(x)dx = f(x+x_0)|_{x=0} = f(x_0)$$

由此导出式（3.4.13）。

对于横坐标的尺度变换，单位冲激函数有一个奇特的性质：

$$\delta(ax) = \frac{1}{|a|}\delta(x) \qquad (3.4.14)$$

式（3.4.14）表明横坐标轴的尺度变换实际上导致了纵坐标的尺度变换。在使用单位冲激函数进行代数操作时要牢记这个性质。证明如下：

证明：令 $f(x)$ 为任一函数，则：

$$\int_{-\infty}^{\infty} \delta(ax)f(x)dx = \frac{1}{|a|}\int_{-\infty}^{\infty} \delta(\sigma)f(\frac{\sigma}{a})d\sigma = \frac{1}{|a|}f(0) \qquad (3.4.15)$$

其中，$ax = \sigma, x = \sigma/a$ 和 $dx = (1/a)d\sigma$。

当 $a < 0$ 时，积分限互换导致了一个负号，故用 $|a|$。

于是：

$$\int_{-\infty}^{\infty} \delta(ax)f(x)dx = \frac{1}{|a|}f(0)$$

$$= \frac{1}{|a|}\int_{-\infty}^{\infty} \delta(x)f(x)dx = \int_{-\infty}^{\infty}\left[\frac{1}{|a|}\delta(x)\right]f(x)dx \qquad (3.4.16)$$

由于 $f(x)$ 为任意函数，故：

$$\delta(ax)=\frac{1}{|a|}\delta(x)$$

当 $a=-1$ 时，$\delta(-x)=\delta(x)$，这里证明 $\delta(x)$ 关于原点对称。

3. 线性系统的冲激响应

有两种办法来表示一个线性移不变系统输入和输出之间的关系：

（1）任何一个这样的系统都有一个复值的传递函数，其与调谐输入相乘就得到对应的调谐输出。

（2）任何一个这样的系统都有一个实数值的冲激响应，其与输入信号的卷积给出对应的输出，其输出称为系统响应。

假设 $\delta(x)$ 为输入信号，$g(x)$ 为系统特征函数，有：

$$\delta(x)*g(x)=\int_{-\infty}^{\infty}\delta(\sigma)g(x-\sigma)d\sigma=g(x-\sigma)\Big|_{\sigma=0}=g(x) \quad (3.4.17)$$

即：表征线性系统行为的特征函数 g(x)是单位冲激函数 $\delta(x)$ 的响应，因此表征线性系统行为的特征函数叫做系统的冲激响应。

还可以得到一个性质：

任何函数与单位冲激函数的卷积是原函数。

这个证明可由式（3.4.13）直接推出。

3.4.5 阶跃函数

阶跃函数是 $x=0$ 处不连续的符号函数 u(x)，定义如下：

$$u(x)=\begin{cases}1 & x>0\\ \frac{1}{2} & x=0\\ 0 & x<0\end{cases} \quad (3.4.18)$$

积分性质：

$$\int_{-\infty}^{\infty}u(x)f(x)dx=\int_{0}^{\infty}f(x)dx \quad (3.4.19)$$

其中 $f(x)$ 为任意函数。图 3-12 给出阶跃函数 $u(x)$ 的波形。

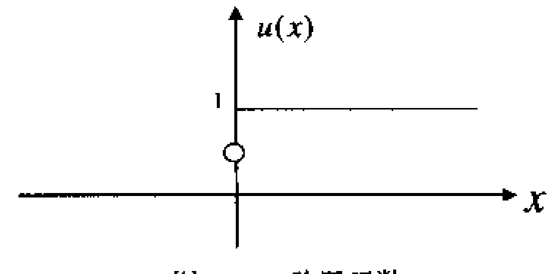


图 3-12 阶跃函数

注意：阶跃函数是单位冲激函数的积分：

$$u(x-x_0)=\int_{-\infty}^{x}\delta(\sigma-x_0)d\sigma=\begin{cases}1, x>x_0\\ 0, x<x_0\end{cases} \quad (3.4.20)$$

相反，可以预料，单位冲激函数是阶跃函数的导数：

$$u'(x)=\frac{du(x)}{dx}=\delta(x) \quad (3.4.21)$$

证明：对任何在 $x=\pm\infty$ 处趋零的函数 f(x)，计算：

$$\int_{-\infty}^{\infty} u'(x)f(x)dx = u(x)f(x)|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} u(x)f'(x)dx$$

$$= -\int_{-\infty}^{\infty} u(x)f'(x)dx$$

$$= -\int_{0}^{\infty} f'(x)dx \qquad \text{（由式（3.4.19））}$$

$$= -[f(\infty) - f(0)]$$

$$= f(0)$$

$$= \int_{-\infty}^{\infty} \delta(x)f(x)dx \qquad \text{（由式（3.4.20））}$$

对任意所选的 $f(x)$ 成立，则（3.4.19）式成立。

3.5 卷积滤波

卷积滤波常用于实现对信号和图像的线性运算。

3.5.1 平滑

矩形脉冲函数 $g(x)$ 通常作为平滑滤波的冲激响应。$f(x)$ 为将被滤波的函数（通常都是被噪声干扰了），卷积时，矩形脉冲从左移到右与 $f(x)$ 做卷积，产生函数 $h(x)$，$h(x)$ 在各点的值等于 $f(x)$ 在单位长度上的局部平均值，这种平均刚好将 $f(x)$ 的噪声平滑掉了，起到去噪的效果。这种局部平均值具有压制波形的高频起伏而保留输入函数 $f(x)$ 的基本波形的作用。

此应用是具有非负冲激响应的滤波器平滑有噪声污染的信号的一个典型例子。也可以用三角函数或高斯脉冲做平滑函数。

$$h(x) = \int_{-\infty}^{\infty} f(\sigma)g(x-\sigma)d\sigma \tag{3.5.1}$$

图 3-13 显示了利用卷积平滑受到干扰的函数 $f(x)$ 时情形。

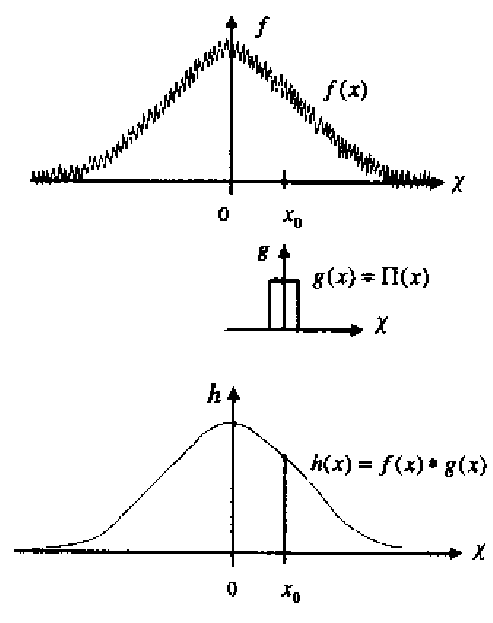


图 3-13 对有噪声函数进行光滑

3.5.2 边缘增强

图 3-14 表示另一种滤波——边缘增强。边缘函数 $f(x)$ 缓慢地从低到高，脉冲响应 $g(x)$ 为一个有着负的旁瓣（Side Lobes）的正尖峰函数。随着卷积的进行 $g(x)$ 从左移到右，旁瓣和主尖峰依次与边缘相遇，输出结果如 $h(x)$ 显示。

图中的边缘增强滤波器具有两方面的影响。首先，它会增加边缘部分的坡度；其次，在边缘渐变部分的两头，它会产生“过冲”或称“加边 ringing”。常用的边缘增强滤波器都有这种现象。

作为第二个边缘增强的例子，考虑如下的冲激响应：

$$g(x) = 2\delta(x) - e^{-x^2/2\delta^2} \qquad (3.5.2)$$

其波形如图 3-15 所示。

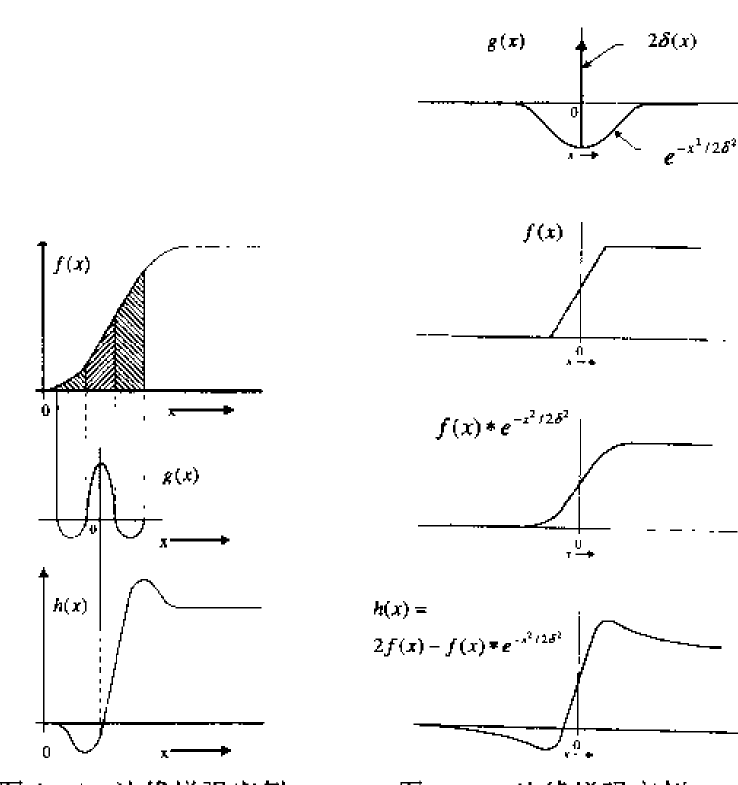


图 3-14 边缘增强实例 1　　图 3-15 边缘增强实例 2

注意：

$$h = f * g = f(x) * 2\delta(x) - f(x) * e^{-x^2/2\delta^2} = 2f(x) - f(x) * e^{-x^2/2\delta^2} \qquad (3.5.3)$$

即输出是两倍输入减去输入和一高斯函数之卷积，与高斯函数的卷积使得边缘变得模糊，但是输出却得到了图中增强的边缘。同样，在此图中也存在着过冲。

这个例子说明从原始图像中减去模糊后的图像具有边缘增强的效果。这种运算可联系到叫做非锐化掩膜（Unsharp Masking）的照相暗室技术。

小结

本章主要介绍了线性系统中的调谐信号与复信号、卷积、一些有用的函数、卷积滤波等内容。

重点：本章重点主要有以下几个方面：

（1）当线性系统的输入是两信号之和时，其输出等于两输入信号单独作用时输出的和。

（2）对于移不变系统，改变输入信号的时空原点只导致输出信号作同样的平移。

（3）由于调谐信号能简化线性系统分析，故常用它来表示正弦信号。

（4）对于线性移不变系统，调谐（正弦型）信号输入产生调谐信号输出。

（5）线性移不变系统的性能可用其传递函数完全刻画。

（6）传递函数是以频率为自变量的负值函数，它反映了调谐信号输入和输出之间幅度和相位的关系。

（7）对线性移不变系统，调谐信号输入乘以对应输入信号频率的传递函数就得到系统的输出。

（8）两个函数的卷积计算将是一个函数反折平移后与另一函数相乘，并计算积函数的积分，卷积的输出就是作为平移距离的函数的积分值。

（9）线性移不变系统的输出等于输入与系统冲激响应函数的卷积。

（10）任一线性移不变系统的冲激响应都是惟一的，并可完全刻画系统。

（11）卷积操作可用作模型描述图像的采样点。

（12）卷积的数字实现可用于对数字信号和图像进行线性滤波。

（13）数字线性滤波可用于去卷积、去噪声和特征增强。

（14）由于卷积和图像边框处不能正确执行，故应避免此处有重要信息。

（15）两个高斯函数的卷积得到一个比原先两个高斯函数都“宽”的高斯函数。

（16）单位冲激 $\delta(x)$ 是在卷积操作下的全同函数。

（17）单位冲激 $\delta(x)$ 横坐标的尺度变化导致其强度的变化。

（18）单位冲激函数是阶跃函数的导数。

（19）边缘增强滤波器的冲激响应一般由原点处的正尖峰与负旁瓣组成。

（20）边缘增强滤波器通常会产生过冲（或称“加边”）。

难点：本章难点在于涉及比较多的数学知识，一些重要的数学推导过程需要掌握，以及理解卷积的定义及在图像处理技术中的应用。

疑点：如何设计一个好的线性系统呢，好的标准又是什么。

练习三

一、选择题

1. 下列（　）是调谐信号。

A. $x(t) = je^{\omega t}$　　B. $x(t) = e^{\alpha t}$

C. $x(t) = \cos(\omega t) + j\sin(\omega t)$　　D. $x(t) = te^{j\omega}$

2. 冲激函数有一个奇特的性质，$\delta(ax) =$（　）。

A. $\frac{1}{|a|}\delta(x)$　　B. $|a|\delta(x)$

C. $a\delta(x)$　　D. $\frac{1}{a}\delta(x)$

3. 以下（　）不属于卷积的性质。

A. $f*g=g*f$

B. $f*(g+h)=f*g+f*h$

C. $f*(g*h)=(f*g)*h$

D. $\frac{d}{dt}(f*g)=f'*g'$

二、思考题

1. 验证式（3.3.15）。
2. 验证式（3.3.16）。
3. 证明 $\Pi(t)*\Pi(t)=\Lambda(t)$。

三、上机题

编写一个程序，求两个向量的卷积。

第 4 章　傅立叶变换

一般数字图像处理的计算方法本质上都可看成为线性的，处理后的输出图像阵列可看为输入图像阵列的各个元素经加权线性组合而得到。这种空间线性处理要比非线性处理简单，但对图像处理运算来说，由于图像阵列很大，如果没有有效的算法，计算上是很麻烦和费时的，往往采用各种图像变换的方法，如：傅立叶变换、沃尔什变换等间接处理技术，可获得更有效的处理。数字图像经二维离散傅立叶变换后，其空间域处理可转换为变换域处理，它具有很多优点，最突出的是算术运算次数将大大减少，并可采用二维数字滤波器技术进行所需要的各种图像处理。

本章的主要内容如下：

（1）傅立叶变换的概念。

（2）傅立叶变换的性质。

（3）线性系统和傅立叶变换。

（4）二维傅立叶变换。

（5）相关性和能量谱的概念。

（6）傅立叶变换性质总结。

4.1　傅立叶变换的概念

法国工程师傅立叶于 1807 年提出了傅立叶级数的概念，即任一周期信号可分解为复正弦信号的叠加。1822 年，傅立叶（Fourier）发表了“热传导解析理论”，提出了傅立叶变换。它本质上提出了一种与空间思维不同的频域思维方法。经过 100 多年的发展，傅立叶变换不但已经形成了一个重要的数学分支，同时也在信号分析与信号处理中起到了重要的作用。正是由于傅立叶变换，原本对人们比较抽象的“频率”概念才变得具体化。

傅立叶变换是十九世纪数学界和工程界最辉煌的成果之一。它一直是信号处理领域中最完美、应用最广泛、效果最好的一种分析手段。它也是线性系统分析的有利工具，它使人们能够定量地分析诸如数字系统、采样点、电子放大器、卷积、滤波、噪声、显示点等的作用。把傅立叶变换理论同其物理解释相结合，将大大有助于解决大多数图像处理问题。

傅立叶变换使人们从空间域（或时域）与频率域两个不同的角度来看待信号或图像的问题。有时在时域无法解决的问题，在频域却是显而易见的。对一个给定的信号，如 $x(t)$，可以用众多的方法来描述它，如 $x(t)$ 的函数表达式，通过傅立叶变换所得到的 $x(t)$ 的频谱，即 $X(j\Omega)$，再如 $x(t)$ 的相关函数，其能量谱或功率谱等。

在这些众多的描述方法中，有两个最基本的物理量，即时间和频率。显然，时间和频率与人们的日常生活关系最为密切，人们时时可以感受到它们的存在。时间自不必说，对频率，如夕阳西下时多变的彩霞，音乐会上那优美动听的旋律以及在一片寂静中突然冒出的一声刺耳的尖叫等，这些都包含了丰富的频率内容。正因为如此，时间和频率也成了描述信号行为的两个最重要的物理量。

信号是变化着的，变化着的信号构成了五彩斑斓的世界。此处所说的“变化”，一是

指信号的幅度随时间变化；二是指信号的频率内容随时间变化。幅度不变的信号是“直流”信号，而频率内容不变的信号是由单频率信号，或多频率信号所组成的信号，如正弦波、方波、三角波等。不论是“直流”信号还是正弦类信号都只携带着最简单的信息。

4.1.1 连续傅立叶变换

函数 $f(x)$ 的一维傅立叶变换定义如下：

$$F\{f(t)\}=F(s)=\int_{-\infty}^{\infty}f(t)e^{-j2\pi st}dt \tag{4.1.1}$$

其中，$j^2=-1$。傅立叶变换是一个线性积分变换，它将一个有 n 个实变量的复函数变换为另一个有 n 个实变量的复函数。$F(s)$ 可以看作 f(t)在谐波 $e^{j2\pi st}$ 上的投影，即 f(t)在频率为 s 的谐波 $e^{j2\pi st}$ 上占有的成份。定义 $F(s)$ 的一维傅立叶反变换为：

$$F^{-1}\{F(s)\}=\int_{-\infty}^{\infty}F(s)e^{j2\pi st}ds \tag{4.1.2}$$

傅立叶积分定理指出：

$$f(t)=\int_{-\infty}^{\infty}[\int_{-\infty}^{\infty}f(t)e^{-j2\pi st}dt]e^{j2\pi st}ds=\int_{-\infty}^{\infty}F(s)e^{j2\pi st}ds \tag{4.1.3}$$

即傅立叶变换是互逆的：

$$F\{f(t)\}=F(s)\Rightarrow F^{-1}\{F(s)\}=f(t) \tag{4.1.4}$$

函数 $f(t)$ 和 $F(s)$ 被称作一个傅立叶变换对。它们是互逆的和惟一决定的：

$$f(t)=F^{-1}\{F(s)\} \quad \Longleftrightarrow \quad F(s)=F\{f(t)\}$$

根据因子 2π在等式中的不同位置，式（4.1.2）、（4.1.3）和（4.1.4）都会有其他的表达方式。这里用到的方式中，频率变量是每个单位 t 的调整周期（而不是弧度）个数来度量的。推广到高维。就 n 维空间而言，s、t 可以看作向量，即：

$$\begin{aligned}s&=(s_1,s_2,s_3,\dots,s_m)\\ t&=(t_1,t_2,t_3,\dots,t_m)\end{aligned} \tag{4.1.5}$$

$s\cdot t$ 可以看作内积，积分看作多重积分，则公式（4.1.2）、（4.1.3）、（4.1.4）仍然成立。

【例 4-1】计算高斯函数的傅立叶变换。

$$f(t)=e^{-\pi t^2} \tag{4.1.6}$$

由（4.1.1）得：

$$\begin{aligned}F(s)&=\int_{-\infty}^{\infty}e^{-\pi t^2}e^{-j2\pi st}dt\\ &=\int_{-\infty}^{\infty}e^{-\pi(t^2+j2st)}dt\\ &=\int_{-\infty}^{\infty}e^{-\pi(t^2+j2st+(js)^2)-\pi s^2}dt\end{aligned} \tag{4.1.7}$$

于是：

$$F(s)=e^{-\pi s^2}\int_{-\infty}^{\infty}e^{-\pi(t+js)^2}du \tag{4.1.8}$$

变量替换：

$$u=t+js \qquad du=dt \tag{4.1.9}$$

则：

$$F(s)=e^{-\pi s^2}\int_{-\infty}^{\infty}e^{-\pi u^2}du \tag{4.1.10}$$

注意到高斯函数的面积为:

$$\int_{-\infty}^{\infty} e^{-x^2/2\sigma^2} dx = \sqrt{2\pi\sigma^2} \qquad (4.1.11)$$

由此可推出:

$$\int_{-\infty}^{\infty} e^{-\pi u^2} du = \int_{-\infty}^{\infty} e^{-u^2/2(\frac{1}{2\pi})} du = 1 \qquad (4.1.12)$$

故，式（4.1.10）为:

$$F(s) = e^{-\pi s^2} \qquad (4.1.13)$$

即高斯函数的傅立叶变换还是高斯函数。这个性质在以后的分析之中非常有用。

4.1.2 傅立叶变换的存在性

傅立叶变换是一个函数与振荡型函数 $e^{-j2\pi st}$ 的乘积在全实数域范围的积分，积分是否存在?

1. 瞬时函数

有一些函数在自变量达到很大的正或负值时足够快地变为零，使得式(4.1.1)和(4.1.2)中的积分存在。如果一个函数 f(t)是绝对可积的函数(f(t)∈L1(R))，即:

$$\int_{-\infty}^{\infty} |f(t)| dt < \infty \qquad (4.1.14)$$

并且函数或者连续，或者只有有限个不连续点，则对∀s∈R，f(t)的傅立叶变换都存在。这些函数被称为瞬时函数，因为在$|t|$很大时函数值已经消失了。

而实际上需要处理的正是这样的函数。任何数字化信号和图像肯定都被截为有限延续和有界的函数。这样将用到的函数都存在傅立叶变换。不过如果能讨论另外一些在严格意义上的变换不存在的函数也是很有用的。

2. 周期函数和恒值函数

对于 $f(t)=\cos(2\pi t)$ 或 $f(t)=1$，对于 s 的所有值，傅立叶变换不存在，但是利用广义函数δ(t)，可获得这两类函数傅立叶变换的有用的表达式。

考虑一对冲激函数的反变换:

$$\begin{aligned} f(t) &= F^{-1}\{\delta(s-f_0)+\delta(s+f_0)\} \\ &= \int_{-\infty}^{\infty} [\delta(s-f_0)+\delta(s+f_0)] e^{j2\pi st} ds \\ &= \int_{-\infty}^{\infty} \delta(s-f_0) e^{j2\pi st} ds + \int_{-\infty}^{\infty} \delta(s+f_0) e^{j2\pi st} ds \\ &= e^{j2\pi f_0 t} + e^{-j2\pi f_0 t} \\ &= 2\cos(2\pi f_0 t) \end{aligned} \qquad (4.1.15)$$

最后一个等式用到了欧拉公式（如下）:

$$\cos\theta = \frac{1}{2}(e^{j\theta}+e^{-j\theta})$$

$$\sin\theta = \frac{1}{2j}(e^{j\theta}-e^{-j\theta}) = -\frac{j}{2}(e^{j\theta}-e^{-j\theta}) \qquad (4.1.16)$$

故:

$$F\{\cos(2\pi f_0 t)\}=\frac{1}{2}[\delta(s+f_0)+\delta(s-f_0)] \tag{4.1.17}$$

即，频率为 f_0 的余弦函数的傅立叶变换是一对分别位于 $s=\pm f_0$ 的脉冲。

同理：

$$F\{\sin(2\pi f_0 t)\}=\frac{j}{2}[\delta(s+f_0)-\delta(s-f_0)] \tag{4.1.18}$$

在式（4.1.17）中，令 $f_0=0$，得：

$$F\{1\}=\delta(s) \tag{4.1.19}$$

即，常数的傅立叶变换是原点的一个脉冲。

现在已经有了关于常数和正弦变换的傅立叶变换的有用表达式。傅立叶级数理论中，任何频率为 f 的周期函数都可以表示为频率为 nf 的正弦型函数的累加，其中 n 取整数值。由加法定理可知周期函数的傅立叶变换是频域内一系列等距冲激。

3. 随机函数

将那些无限延伸的、非常数、非周期、绝对积分不存在的函数归并为一类，统称为随机函数。通常要求随机函数的自相关函数是存在的，即：

$$R_f(\tau)=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T}f(t)f(t+\tau)dt<\infty \tag{4.1.20}$$

许多函数，它们的自相关函数都是存在的。自相关函数是偶函数，它们的傅立叶变换是 $f(t)$ 的能量谱，以后将会介绍这些概念。

在这种情况下，只要重新定义傅立叶变换，则积分存在：

$$F(s)=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T}f(t)e^{-j2\pi st}dt \tag{4.1.21}$$

其中：

$$f(t)=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T}F(s)e^{j2\pi st}ds \tag{4.1.22}$$

这样就可以处理重新定义其变换的函数了。

4.1.3 傅立叶级数的展开

设 f(t)为周期为 T 的信号，傅立叶级数展开式为：

$$f(t)=\frac{a_0}{2}+\sum_{n=1}^{\infty}a_n\cos(2\pi\frac{n}{T}t)+\sum_{n=1}^{\infty}b_n\sin(2\pi\frac{n}{T}t) \tag{4.1.23}$$

其中：

$$a_n=\frac{2}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}}f(x)\cos(2\pi\frac{n}{T}x)dx$$

$$b_n=\frac{2}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}}f(x)\sin(2\pi\frac{n}{T}x)dx \tag{4.1.24}$$

两个无限实系数系列{ a_n }和{ b_n }表示了一个周期为 T 的函数。

提示：设周期函数 $f(t)$，$t\in[-\frac{T}{2},\frac{T}{2}]$。

则傅立叶反变换为：

$$f(t)=\int_{-\infty}^{\infty}F(s)e^{j2\pi st}ds \tag{4.1.25}$$

其中：

$$F(s) \approx \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t) e^{-j2\pi st} dt \quad (4.1.26)$$

将频域分割，令 $\Delta s = \frac{1}{T}$，这时：

$$f(t) \approx \sum_{n=-\infty}^{\infty} F(n\Delta s) e^{j2\pi n\Delta st} \Delta s = \frac{1}{T} \sum_{n=-\infty}^{\infty} F(n\Delta s) e^{j2\pi \frac{n}{T} t} \quad (4.1.27)$$

其中：

$$F(n\Delta s) = \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t) e^{-j2\pi n\Delta st} dt = \int_{-\frac{T}{2}}^{\frac{T}{2}} f(t) e^{-j2\pi \frac{n}{T} t} dt \quad (4.1.28)$$

注意到：

$$e^{j2\pi st} = \cos(2\pi st) + j\sin(2\pi st) \quad (4.1.29)$$

可得到式（4.1.23）、（4.1.24）。即用无限实系数系列表示了一个周期为 T 的函数。

4.1.4 离散傅立叶变换

令 g(t)为瞬时函数或周期函数的一个周期，满足：

$g(t) = 0$，　当 $|t| > \frac{T}{2}$ 时

并且假设 g(t)最高截止频率是有限的，即：

$G(s) = 0$，　当 $|s| > S_m$ 时，

其中：

$$G(s) = G\{g(t)\} = \int_{-\infty}^{\infty} g(t) e^{-j2\pi st} dt \quad (4.1.30)$$

因为 $g(t)$ 和 $G(s)$ 都为周期函数（或可拓展为周期函数），观察主周期：

$g(t)$，$t \in [-\frac{T}{2}, \frac{T}{2}]$

$G(s)$，$s \in [-S_m, S_m]$

（1）将时间和频率都离散化。

若将 $[-\frac{T}{2}, \frac{T}{2}]$ 细分为 N 份，取 $\Delta t = \frac{T}{N}$，则 $S_m = \frac{1}{2\Delta t}$。这是说如果将 $[-S_m, S_m]$ 也细分为 N 份，则 $\Delta s = \frac{1}{T}$，$\Delta s \Delta t = \frac{1}{N}$。

（2）频率和时间离散化，这时：

$$G_n = G(n\Delta s) = \int_{-\frac{T}{2}}^{\frac{T}{2}} g(t) e^{-j2\pi(n\Delta st)} dt \approx \sum_{i=0}^{N-1} g(i\Delta t) e^{-j2\pi(n\Delta s)i\Delta t} \Delta t$$
$$= \frac{T}{N} \sum_{i=0}^{N-1} g_i e^{-j2\pi(\frac{n}{N})i} \quad (4.1.31)$$

其中，n=0,1,…,N-1。

$$g_i = g(i\Delta t) = \int_{-S_m}^{S_m} G(s) e^{j2\pi(i\Delta t)s} ds \approx \sum_{i=0}^{N-1} G(n\Delta s) e^{j2\pi(i\Delta t)(n\Delta s)} \Delta s$$
$$= \frac{1}{T} \sum_{n=0}^{N-1} G_n e^{j2\pi(\frac{i}{N})n} \quad (4.1.32)$$

其中，i = 0,1,..., N-1。

（3）如果选择 $T=\sqrt{N}$，则 $\frac{T}{N}=\frac{1}{T}=\frac{1}{\sqrt{N}}$。

如果 $\{f_i\}$ 是一个长度为 N 的序列，则离散傅立叶变换（DFT）为序列 $\{F_n\}$：

$$F_n=\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}f_i e^{-j2\pi\frac{n}{N}i} \qquad (4.1.33)$$

而逆 DFT 为：

$$f_i=\frac{1}{\sqrt{N}}\sum_{n=0}^{N-1}F_n e^{j2\pi\frac{i}{N}n} \qquad (4.1.34)$$

其中：

$0 \leqslant i,n \leqslant N-1$

离散傅立叶变换具有连续傅立叶的性质。对于在数字图像处理中遇到的函数类型，两者相差的确很小，事实上如果 $\{f_i\}$ 是由某一常用类型的连续函数经过适当采样获得的，DFT 可以看作是连续傅立叶变换的一个特例。适当采样这些所谓有限带宽函数，然后用 DFT 计算其傅立叶变换。

DFT 和连续傅立叶变换有如此相似，所以只要遵守采样定理，就可以认为它们完全是等同的。这种灵活性使人们在设计过程中有了相当大的自由度。例如对于一个图像处理问题，可以用连续方法来描述它，然后用离散的方法来实现。

4.1.5 快速傅立叶变换（FFT）

要计算采样信号或图像的傅立叶变换时，通常用 DFT 来实现。实现式（4.1.33）和（4.1.34）所需的乘法和加法操作次数显然是 N^2，即使所有的复指数值都存在一张表中。这样计算量也实在太大了。幸运的是存在一类算法可以将操作降低到（$N\log_2(N)$）数量级，这就是所谓的快速傅立叶变换（FFT）算法。N 必须可以分解为一些较小的整数的乘积，当 N 是 2 的幂时（即 $N=2^p$，其中 p 是整数），效率最高，实现起来也最简单。

注意式（4.1.33）可以写成矩阵形式相乘：

$$\begin{bmatrix} F_0 \\ .. \\ .. \\ F_{N-1} \end{bmatrix} = \begin{bmatrix} W_{0,0} & W_{0,1} & \cdots\cdots & W_{0,N-1} \\ W_{1,0} & W_{1,1} & \cdots\cdots & W_{1,N-1} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ W_{N-1,0} & W_{N-1,1} & \cdots\cdots & W_{N-1,N-1} \end{bmatrix} * \begin{bmatrix} f_0 \\ .. \\ .. \\ f_{N-1} \end{bmatrix} \qquad (4.1.35)$$

或者：

$$F=\varpi f \qquad (4.1.36)$$

其中：

$$\varpi_{n,i}=\frac{1}{\sqrt{N}}e^{-j2\pi\frac{ni}{N}} \qquad (4.1.37)$$

幂函数由于乘积 ni 而具有周期性，矩阵 ϖ 有很好的对称性。矩阵可被分解为包含多个重复值（其中还有许多 0 和 1）的 $N\times N$ 矩阵相乘。

如果 $N=2^p$，则 ϖ 可被分解为 p 个这样的矩阵。实现 p 个这样的矩阵相乘所需的操作远小于用式（4.1.33）计算所需的操作。

用 FFT 计算量的减小因子为：

$$\frac{N^2}{N\log_2(N)}=\frac{N}{\log_2(N)} \qquad (4.1.38)$$

这个值随 N 增长而增大，当 $N=1024$ 时，FFT 比直接计算降了 100 倍。

可以参考：

（1）夏德深等，现代图像处理技术及其应用，东南大学出版社。

（2）MVWickerhauser.Adapted wavelet analysis from theory to software.1994:P69。

4.1.6　一些常用的傅立叶变换

表 4-1 给出了一些有用的常见函数的傅立叶变换。

表 4-1　一些常用函数的傅立叶变换

函数	$f(t)$	$F(s)$
高斯	$e^{-\pi t^2}$	$e^{-\pi s^2}$
矩形脉冲	$\Pi(t)$	$\frac{\sin(\pi s)}{\pi s}$
三角脉冲	$\Lambda(t)$	$\frac{\sin^2(\pi s)}{(\pi s)^2}$
冲激	$\delta(t)$	1
单位阶跃	$u(t)$	$\frac{1}{2}[\delta(s)-\frac{j}{\pi s}]$
余弦	$\cos(2\pi ft)$	$\frac{1}{2}[\delta(s+f)+\delta(s-f)]$
正弦	$\sin(2\pi ft)$	a
复指数	$e^{j2\pi ft}$	$\delta(s-f)$

4.2　傅立叶变换的性质

4.2.1　对称性

一般情况下，一个单实变量负值函数的傅立叶变换也是一个单实变量负值函数。但是也存在几类有限的函数，由于它们在傅立叶变换下所表现出的对称性，使之显得尤为引人注目。

1.　奇偶性

函数 $f_e(t)$ 为偶函数当且仅当：

$$f_e(t)=f_e(-t) \qquad (4.2.1)$$

函数 $f_e(t)$ 为奇函数当且仅当：

$$f_o(t)=-f_o(-t) \qquad (4.2.2)$$

任意函数 $f(x)$ 可分为奇、偶两部分组成，只需令：

$$f_e(t)=\frac{1}{2}[f(t)+f(-t)] \qquad (4.2.3)$$

和：

$$f_o(t)=\frac{1}{2}[f(t)-f(-t)] \qquad (4.2.4)$$

这里：

$$f(t)=f_e(t)+f_o(t) \tag{4.2.5}$$

考虑奇偶性对傅立叶变换的影响。由欧拉关系：

$$e^{jx}=\cos(x)+j\sin(x) \tag{4.2.6}$$

$a=\dfrac{1}{\sqrt{2\pi\sigma^2}}$ 的傅立叶变换可变为：

$$F(s)=\int_{-\infty}^{\infty}f(t)e^{-j2\pi st}dt=\int_{-\infty}^{\infty}f(t)\cos(2\pi st)dt-j\int_{-\infty}^{\infty}f(t)\sin(2\pi st)dt \tag{4.2.7}$$

将 $f(t)$ 写成奇偶两部分的和的形式，则有：

$$\begin{aligned}F(s)=\int_{-\infty}^{\infty}f(t)e^{-j2\pi st}dt&=\int_{-\infty}^{\infty}f_e(t)\cos(2\pi st)dt+\int_{-\infty}^{\infty}f_o(t)\cos(2\pi st)dt\\&-j\int_{-\infty}^{\infty}f_e(t)\sin(2\pi st)dt-j\int_{-\infty}^{\infty}f_o(t)\sin(2\pi st)dt\end{aligned} \tag{4.2.8}$$

注意第二、三项，奇函数和偶函数之积为奇函数，其无穷积分为零从而傅立叶变换可以简化为如下的形式：

$$\begin{aligned}F(s)&=\int_{-\infty}^{\infty}f_e(t)\cos(2\pi st)dt-j\int_{-\infty}^{\infty}f_o(t)\sin(2\pi st)dt\\&=F_e(s)+jF_o(s)\end{aligned} \tag{4.2.9}$$

总结傅立叶变换的对称性：

（1）偶函数分量的变换为偶函数分量。

（2）奇函数分量的变换为奇函数分量。

（3）奇函数分量引入系数 $-j$。

（4）偶函数分量不引入系数。

2. 实部和虚部

利用上述四种规则可以推导出傅立叶变换对复函数的作用。如果将一个复函数分成四部分的和——奇和偶的实部，奇和偶的虚部——就可以列出以下四条傅立叶变换规则：

（1）实的偶部产生实的偶部。

（2）实的奇部产生虚的奇部。

（3）虚的偶部产生虚的偶部。

（4）虚的奇部产生实的奇部。

由于人们通常用实函数来表示输入图像，因此研究输入函数为实函数的情况尤为重要。注意实函数的变换结果具有偶实部和奇虚部，于是有如下定义：

定义 1： Hermite 函数是具有实偶部和奇虚部的函数。它具有共轭对称性。

定义 2： 共轭对称性：

$$F(s)=F^*(-s) \tag{4.2.10}$$

其中*表示负数的共轭。

性质： 实函数的傅立叶变换是 Hermite 函数，具有共轭对称性。

$$F(s)=F_e(s)+jF_o(s)$$

$$F(-s)=\int_{-\infty}^{\infty} f(t)e^{-j2\pi(-s)t}dt=\overline{F(s)}$$

傅立叶变换的所有对称性都在表 4-2 中列出。注意反变换和正变换仅在奇部的符号上有所不同，这意味着偶函数的正、反变换是一样的。

表 4-2　傅立叶变换的对称性

$f(t)$	$F(s)$
偶	偶
奇	奇
实偶	实偶
实奇	虚奇
虚偶	虚偶
复偶	复偶
复奇	复奇
实	Hermite
虚	反 Hermite
实偶，加虚奇	实
实奇，加虚偶	虚

4.2.2　加法定理

假设有两个傅立叶变换对：

$$F\{f(t)\}=F(s) \tag{4.2.11}$$

和：

$$F\{g(t)\}=G(s) \tag{4.2.12}$$

如果将两个时间函数相加，则它们之和的傅立叶变换为：

$$F\{f(t)+g(t)\}=\int_{-\infty}^{+\infty}[f(t)+g(t)]e^{-j2\pi st}dt \tag{4.2.13}$$

经整理得：

$$F\{f(t)+g(t)\}=\int_{-\infty}^{+\infty}f(t)e^{-j2\pi st}dt+\int_{-\infty}^{+\infty}g(t)e^{-j2\pi st}dt=F(s)+G(s) \tag{4.2.14}$$

即时域和空间域内的相加对应于频域内的相加，如图 4-1 所示。这和系统中的线性概念相吻合。

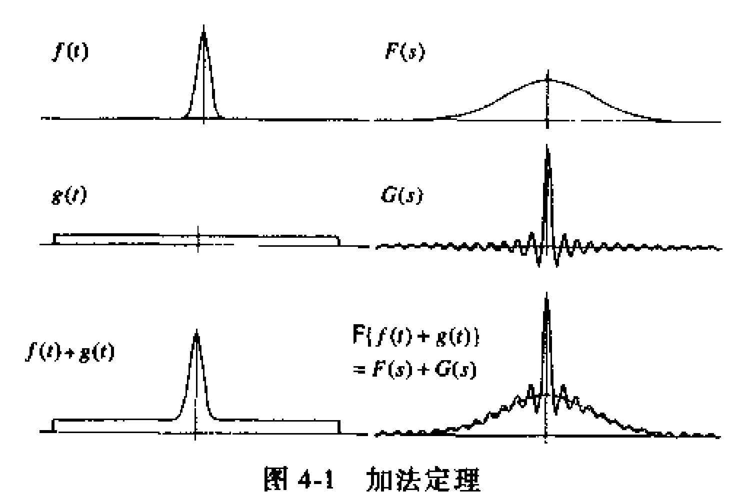


图 4-1　加法定理

由加法定理可得：

$$F\{a_1 f(t)+a_2 g(t)\}=a_1 F\{f(t)\}+a_2 F\{g(t)\} \qquad (4.2.15)$$

其中 a_1 和 a_2 是有理常数。式（4.2.15）适用于任何常数作为一个公理。

4.2.3 位移定理

位移定理描述了移动一个函数的原点对变换的影响。定理如下：

$$F\{f(t-a)\}=e^{-j2\pi as}F(s) \qquad (4.2.16)$$

$$\left|F\{f(t-a)\}\right|=\left|F\{f(t)\}\right|$$

其中，$|\bullet|$ 为模运算，取幅值。

证明：

$$\begin{aligned}F\{f(t-a)\}&=\int_{-\infty}^{\infty} f(t-a)e^{-j2\pi st}dt\\&=\int_{-\infty}^{\infty} f(u)e^{-j2\pi s(u+a)}du \qquad (令u=t-a)\\&=e^{-j2\pi as}\int_{-\infty}^{\infty} f(u)e^{-j2\pi su}du=e^{-j2\pi as}F(s)\end{aligned} \qquad (4.2.17)$$

因此，函数的位移会在傅立叶变换中引入一项复系数，注意如果 $a=0$，则系数为单位1。复系数：

$$e^{-j2\pi as}=\cos(2\pi as)-j\sin(2\pi as) \qquad (4.2.18)$$

具有单位幅值，在复平面的旋转角随着 s 增加而改变。这说明函数位移不改变其傅立叶变换的幅值（模），但是改变了实部和虚部之间的能量分布，结果是产生一个与频率 s 和位移量 a 均成正比的相移。

4.2.4 卷积定理

卷积定理是线性系统分析中最重要的一条定理。简单来说，在一个域中的卷积，在另一个域中对应的运算会是什么呢？于是有如下的结论：

$$F\{f(t)*g(t)\}=F(s)G(s) \qquad (4.2.19)$$

$$F^{-1}\{F(s)G(s)\}=f(t)*g(t) \qquad (4.2.20)$$

这意味着在一个域（时域）中的卷积相当于另一个域（频域）中的相乘，同样在频域中相乘又相当于在时域中做卷积。

证明：由定义：

$$\begin{aligned}F\{f(t)*g(t)\}&=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(u)g(t-u)du e^{-j2\pi st}dt\\&=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(u)g(t-u)e^{-j2\pi s(t-u)-j2\pi su}dtdu\\&=\int_{-\infty}^{+\infty} g(v)e^{-j2\pi s(v)}dv\cdot\int_{-\infty}^{+\infty} f(u)^{-j2\pi su}du \qquad (令v=t-u)\\&=F(s)G(s)\end{aligned} \qquad (4.2.21)$$

卷积定理指出了傅立叶变换的一个主要好处：与其在一个域中作不直观的、难懂的卷积，不如在另一个域中作简单的乘法，可以达到相同的效果。

可以用卷积定理推导出冲激函数的傅立叶变换。回顾：

$$f(t)*\delta(t)=f(t) \tag{4.2.22}$$

就是说，任何函数和冲激函数卷积后不变。由卷积定理：

$$F(s)F\{\delta(t)\}=F(s) \tag{4.2.23}$$

因为上式对任意函数 $f(t)$ 成立，所以可以选一个 $F(s)$ 没有零值的函数（例如高斯函数）；然后将等式两边同时除以 $F(s)$，得：

$$F\{\delta(t)\}=1 \tag{4.2.24}$$

由此证明冲激函数的傅立叶变换是单位 1。

4.2.5　相似性定理

相似性定理描述了函数自变量的尺度变化对其傅立叶变换的作用。

改变自变量的尺度会将一个函数展宽或压缩。例如在式（4.1.1）中的自变量前乘上一个比例系数可以使得函数伸长或压缩，这时它的傅立叶变换变为：

$$F\{f(at)\}=\int_{-\infty}^{+\infty} f(at)e^{-j2\pi st}dt \tag{4.2.25}$$

将积分和指数都乘以 $1/a$，得：

$$F\{f(at)\}=\frac{1}{a}\int_{-\infty}^{+\infty} f(at)e^{-j2\pi st}dat \tag{4.2.26}$$

作变量替换：

$$u=at \qquad du=adt \tag{4.2.27}$$

可得：

$$F\{f(at)\}=\frac{1}{|a|}\int_{-\infty}^{+\infty} f(u)e^{-j2\pi at(s/a)}du \tag{4.2.28}$$

即得到相似性定理如下：

$$F\{f(at)\}=\frac{1}{|a|}F(\frac{s}{a}) \tag{4.2.29}$$

如果系数 a 大于 1，函数 $f(t)$ 在水平方向收缩，由式（4.2.29）可知傅立叶变换的幅值将缩小 a 倍，同时在水平方向扩展 a 倍。a 小于 1 时作用相反，如图 4-2 所示。

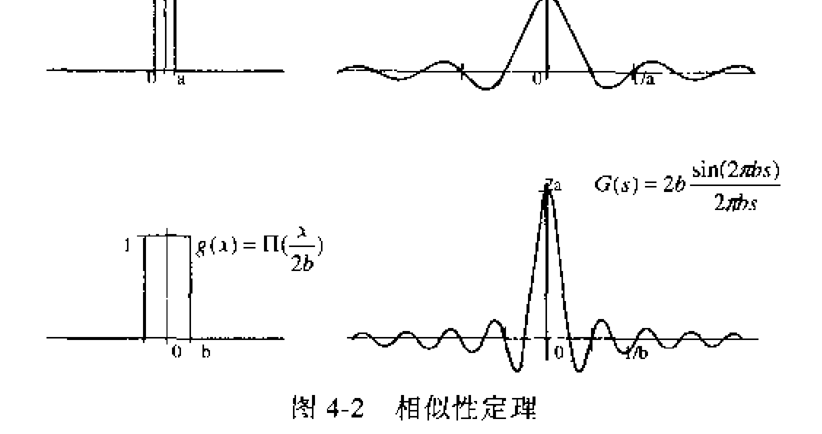


图 4-2　相似性定理

相似性定理意味着一个“窄”函数有一个“宽”的傅立叶变换，反之亦然。可用相似性定理推导出高斯函数的傅立叶变换的一般表达式。高斯函数的傅立叶变换也是高斯函数：

$$F\{e^{-\pi t^2}\}=e^{-\pi s^2} \tag{4.2.30}$$

由相似性定理：

$$F\{e^{-\pi(at)^2}\}=\frac{1}{a}e^{-\pi(s/a)^2} \tag{4.2.31}$$

令：

$$e^{-\pi(at)^2}=e^{-t^2/2\sigma^2} \tag{4.2.32}$$

其中：

$$a=\frac{1}{\sqrt{2\pi\sigma^2}} \tag{4.2.33}$$

于是变换可表示为：

$$F\{e^{-\pi(at)^2}\}=\sqrt{2\pi\sigma^2}\,e^{-2\pi^2\sigma^2 s^2} \tag{4.2.34}$$

由于它也是一个高斯函数，可以定义其标准偏差为α，则：

$$e^{-2\pi^2\sigma^2 s^2}=e^{-s^2/2\alpha^2} \tag{4.2.35}$$

也就是：

$$2\pi^2\sigma^2 s^2=\frac{s^2}{2\alpha^2} \tag{4.2.36}$$

或：

$$\alpha=\frac{1}{2\pi\sigma} \tag{4.2.37}$$

所以对于标准偏差等于σ的高斯函数，其傅立叶变换为：

$$F\{e^{-\pi(at)^2}\}=\sqrt{2\pi\sigma^2}\,e^{-2\pi^2\sigma^2 s^2}\qquad \alpha=\frac{1}{2\pi\sigma} \tag{4.2.38}$$

因此，对于一个标准偏差为σ的单位幅值高斯函数，它的傅立叶变换也是一个高斯函数，幅值为$\sqrt{2\pi\sigma^2}$，标准差为$1/(2\pi\sigma)$。

可以用相似性定理再次说明冲激函数的傅立叶变换是常数。

假设：

$$f(t)=ae^{-\pi(at)^2} \tag{4.2.39}$$

其傅立叶变换为：

$$F(s)=e^{-\pi(s/a)^2} \tag{4.2.40}$$

当a趋近于无穷时，$f(t)$变窄，其幅值增长，同时$F(s)$拓宽，幅值接近常数 1。因此在极限的情况下，高斯函数收缩为脉冲，而其傅立叶变换（拓宽的高斯函数）则趋近于 1，即：$a\to\infty$时，$f(x)\to\delta(x)$，　$F(s)\to 1$。

4.2.6 Rayleigh 定理

一类重要的函数是仅在有限区间内非零的函数，对于这一类的函数，可以讨论其总能量。

函数的能量定义为：

$$\text{energy} = \int_{-\infty}^{+\infty} |f(t)|^2 dt \qquad (4.2.41)$$

条件是积分必须存在。对于瞬时函数，式（4.2.37）存在，并且能量是一个能反映函数总的“大小”的参数。

Rayleigh 定理如下：

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |F(s)|^2 ds \qquad (4.2.42)$$

即变换函数与原函数有着相同的能量。说明信号的能量不论是在时域还是频域都是一样大的，即能量守恒。

Rayleigh 定理的证明如下。

首先有：

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} f(t)f^*(t)dt = \int_{-\infty}^{\infty} f(t)f^*(t)e^{j2\pi ut} dt \qquad u=0 \qquad (4.2.43)$$

即当 $u=0$ 时第二个等号成立。由于 $f(t)$ 通常都是复函数，用 $*$ 表示 $f(t)$ 的共轭函数 $f^*(t)$。将式（4.2.39）看作在频率 $u=0$ 处取值的两个函数乘积的傅立叶反变换。

由于：

$$F^{-1}\{f(t)f^*(t)\} = F(u) * F^*(-u) \qquad u=0 \qquad (4.2.44)$$

将卷积积分写成如下形式：

$$F^{-1}\{f(t)f^*(t)\} = \int_{-\infty}^{\infty} F(s)F^*(s-u)ds \qquad u=0 \qquad (4.2.45)$$

将 $u=0$ 代入式（4.2.45），得到：

$$F^{-1}\{f(t)f^*(t)\} = \int_{-\infty}^{\infty} F(s)F^*(s)ds \qquad u=0 \qquad (4.2.46)$$

即证明了式（4.2.42），说明在两个域中能量相同。如果 $f(t)$ 是实偶函数，则 $F(s)$ 也是实偶函数，于是：

$$\int_{-\infty}^{\infty} f^2(t)dt = \int_{-\infty}^{\infty} F^2(s)ds \qquad (4.2.47)$$

注意：Rayleigh 定理与相似性定理是一致的；如果函数变窄，但是幅值不变，显然它的能量将降低；相似性定理指出，压缩一个函数相当于拓宽其变换，同时也缩减其幅值，从而保证在两个域中能量相等。

4.3 线性系统和傅立叶变换

这一节将分析傅立叶变换在线性系统分析中所起的重要作用。

4.3.1 线性系统术语

如图 4-3 所示为线性系统的常用术语。

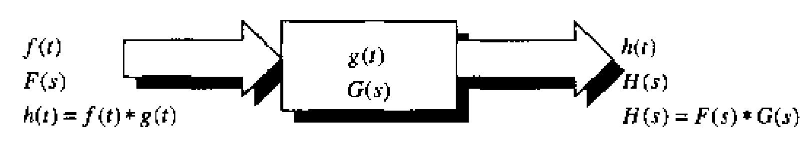


图 4-3 线性系统术语

（1）信号的傅立叶变换称为信号的谱。

（2）谱的傅立叶反变换称为信号。

（3）冲激响应 g(t)与传递函数 G(S)是一对傅立叶变换。

$f(t)$ = 输出信号； $f(s)$ = 输入信号的谱

$g(t)$ = 冲激响应； $G(s)$ = 传递函数

$h(t)$ = 输出信号； $H(s)$ = 输出信号的谱

4.3.2 线性系统辨识

人们通常不知道系统的冲激响应以及其传递函数，需要对其加以确定。确定线性系统中的特征函数 g(t)或 G(s)的过程，称为系统辨识（System Identification）。对于图 4-3 所示的线性系统，由卷积定理可知：

$$H(s)=F(s)G(s) \tag{4.3.1}$$

由此可得：

$$G(s)=\frac{H(s)}{F(s)} \qquad F(s)\neq 0 \tag{4.3.2}$$

因此：

$$g(t)=F^{-1}\left\{\frac{F\{h(t)\}}{F\{f(t)\}}\right\} \tag{4.3.3}$$

这就是说，可以输入一个已知的 $f(t)$，测量其输出 $h(t)$，然后通过数字积分来计算 $g(t)$。例如，假定 $f(t)$ 是一个冲激函数，则 $g(t)$ 就是冲激响应，这样就可直接辨识系统。

下面看一个有趣的例子。

【例 4-2】系统识别。

假定输入为：

$$f(t)=\Pi(t) \qquad F\{f(t)\}=\frac{\sin(\pi s)}{\pi s} \tag{4.3.4}$$

如图 4-4 所示，测得系统输出为：

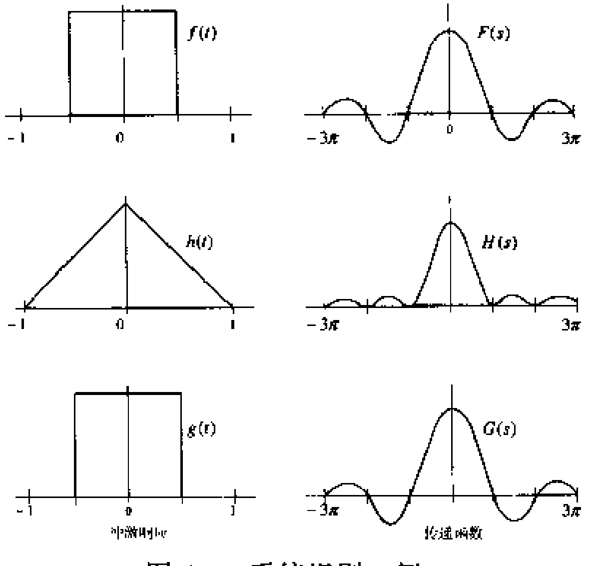


图 4-4 系统识别，例 1

$$h(t)=\Lambda(t) \qquad F\{h(t)\}=\frac{\sin^2(\pi s)}{(\pi s)^2} \tag{4.3.5}$$

则冲激响应为：

$$g(t)=F^{-1}\left\{\frac{\frac{\sin^2(\pi s)}{(\pi s)^2}}{\frac{\sin(\pi s)}{\pi s}}\right\}=F^{-1}\left\{\frac{\sin(\pi s)}{\pi s}\right\}=\Pi(t) \tag{4.3.6}$$

【例 4-3】 线性系统识别。

如图 4-5 所示，输入为：

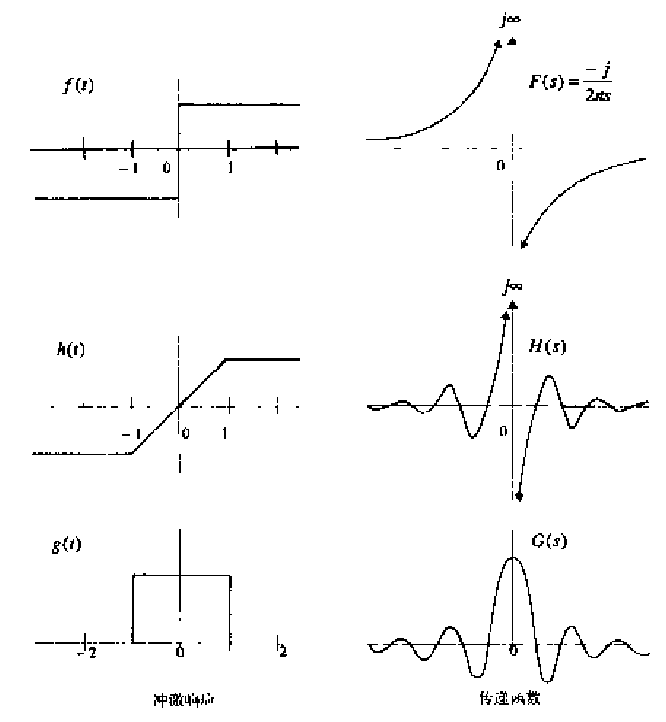


图 4-5　线性系统识别，例 2

$$f(t)=u(t)-\frac{1}{2}=\begin{cases}-\frac{1}{2}, & t<0\\ 0, & t=0\\ +\frac{1}{2}, & t>0\end{cases} \tag{4.3.7}$$

这是一个阶跃函数，它的谱为：

$$F(s)=\frac{-j}{2\pi s} \tag{4.3.8}$$

若系统响应为：

$$h(t)=\begin{cases}-\frac{1}{2}, & t<-1\\ t, & -1\leqslant t\leqslant 1\\ +\frac{1}{2}, & t>1\end{cases} \tag{4.3.9}$$

则此响应的谱为：

$$H(s)=-j\frac{\sin(\pi s)}{2(\pi s)^2} \tag{4.3.10}$$

则由此可以得到线性系统的传递函数如下：

$$G(s)=\frac{H(s)}{F(s)}=\frac{\sin(\pi s)}{\pi s} \tag{4.3.11}$$

通过傅立叶反变换又可以得到冲激响应：

$$g(t)=\Pi(t) \tag{4.3.12}$$

在以上两个例子中，系统输出是解析函数。函数 $f(t)$ 在区域 D 内每一点可微，则称函数 $f(t)$ 在区域 D 内解析；如果 $f(t)$ 在 t_0 的一个邻域内解析，则说 $f(t)$ 在 t_0 点解析，线性系统的辨识问题可以直接解决了。

但是在通常情况下，辨识过程是这样的：输出首先被数字化，通过数字积分变换作用输入和输出，然后直接计算式（4.3.11）的比值，最后在通过傅立叶反变换得到式（4.3.12）。一般都利用快速傅立叶变换（FFT）。

当选择输入函数时，必须注意，其谱中不能有零值。如果有零值，式（4.3.11）中的比值会在某点没有意义。

但是在例 4-2 中选择的输入函数（如图 4-5）违反了这条约束，但是在计算过程中并没有产生错误，其原因是刚好输出函数的谱在相应的点的值也为零。所以在数字反变换之前，$G(s)$ 可以用一些周围的点值进行插值处理。

4.3.3 正弦型分解

傅立叶变换是一个线性积分变换，它以虚指数函数作为其核函数。如式（4.2.7）所示，傅立叶变换可以被表示为两个分别以正弦和余弦函数为核的变换之和，因为 $e^{j2\pi st}$ 可分解为 $\cos(2\pi st)+j\sin(2\pi s)$。因此在傅立叶变换下，正弦和余弦函数会表现出特殊的性质就不足为怪了。

为了对线性系统的冲激响应及其传递函数之间的关系有更深入的了解，需要考虑如图 4-6 所示的线性系统，为了方便，假定 $f(t)$ 和 $g(t)$ 是实偶函数。图 4-6 中两个域画出了输入及其冲激响应和它们的谱。

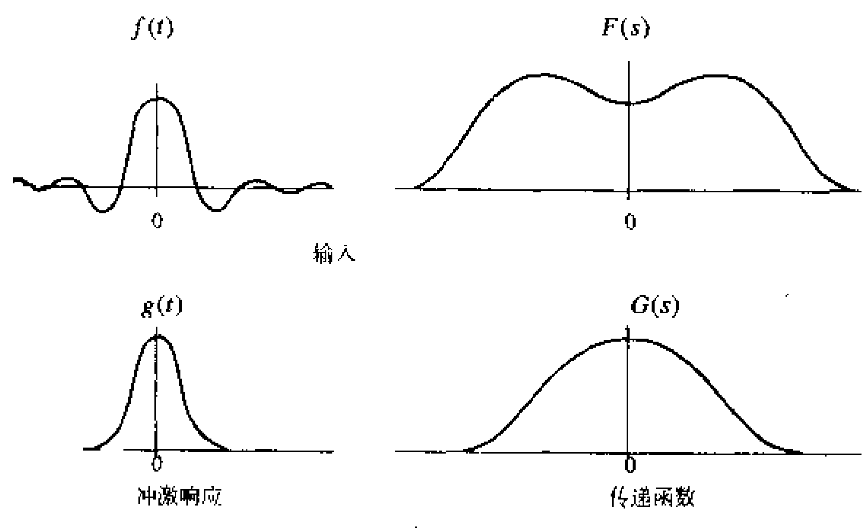


图 4-6 线性系统示例

在输入谱中，以等间隔 Δs 划分 s 轴，并将 $F(s)$ 分解为宽度为 Δs 的一些小长条。如果 Δs 足够小，则 $F(s)$ 可以很好地近似为矩形脉冲的和，如图 4-7 所示。

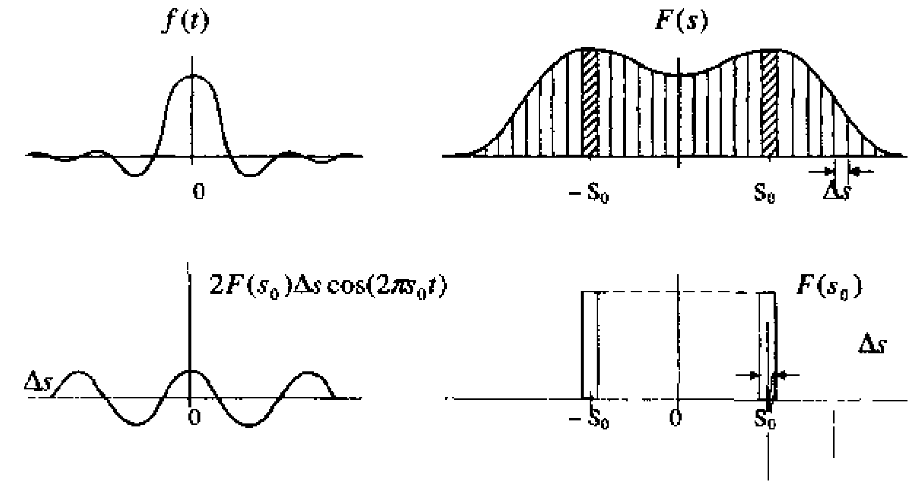


图 4-7 正弦型分解

注意： $F(s)$ 可以用无穷多个如下的脉冲对之和作为一种近似：

$$F(s) \approx \sum_{i=1}^{\infty} F(i\Delta s)\left[\Pi\left(\frac{s-i\Delta s}{\Delta s}\right)+\Pi\left(\frac{s+i\Delta s}{\Delta s}\right)\right] \tag{4.3.13}$$

考虑其中一个特定的位于 $s=\pm s_0$ 处的脉冲对，其幅值为 $F(s_0)$，宽度为 Δs，面积为 $F(s_0)\Delta s$。当 Δs 趋近于零时，在 $s=\pm s_0$ 处的脉冲对为一个偶冲激对，其冲激强度为 $F(s_0)\Delta s$ 为无限小。这个偶冲激对的傅立叶变换趋近于：

$$2F(s_0)\Delta s\cos(2\pi s_0 t) \qquad s_0 = i\Delta s \tag{4.3.14}$$

由于 $F(s)$ 趋近于偶脉冲对的和，如式（4.3.13），则 $f(t)$ 趋近于形如式（4.3.14）的余弦函数的和。这说明任何的偶函数都可以被分解为无穷个幅值无限小的余弦函数的和。

由于输出谱是输入谱和传递函数的乘积，如式（4.3.1），输出信号可被表示成如下形式的余弦函数的和：

$$2G(s_0)F(s_0)\Delta s\cos(2\pi s_0 t) \tag{4.3.15}$$

由上面的准备知识，可以得到整个过程的简单流程：

首先由线性系统的原理可知：

$$h(t) = f(t) * g(t) \tag{4.3.16}$$

所以通过反变换得到：

$$h(t) = F^{-1}\{F(s)G(s)\} \tag{4.3.17}$$

由前面的讨论，可以推导出：

$$f(t) \approx \sum_{s_0=1}^{\infty} 2F(s_0)\Delta s\cos(2\pi s_0 t) \tag{4.3.18}$$

$$h(t) \approx \sum_{s_0=1}^{\infty} 2F(s_0)G(s_0)\Delta s\cos(2\pi s_0 t) \tag{4.3.19}$$

由此就得到了输出信号 $h(t)$ 的估计。

现在可以这样理解线性滤波器：一个输入信号 $f(t)$ 首先被分解为不同频率的余弦函数的和，每个余弦函数的幅值由 $F(s)$ 惟一确定；而 $F(s)$ 是 $f(t)$ 的傅立叶变换，并且是惟一的。

在线性系统中，每个频率为 s_0 的余弦函数都乘以传递函数在该频率处的幅值 $G(s_0)$，然后将所有经过幅度调整后的余弦函数在滤波器的输出端相加，从而得到输出信号 $h(t)$。这里要注意的是，这种解释同前面讨论过的线性系统的两个性质保持一致：

（1）正弦型输出通常产生相同频率的正弦型输出。

（2）传递函数在频率 s 处的值就是频率为 s 的输入正弦型函数幅值被改变的倍数。

由上一节中的傅立叶变换的性质可知，若输入 $f(t)$ 是奇函数，则它的傅立叶变换 $F(s)$ 是虚奇函数，脉冲对也是虚奇函数，$f(t)$ 将被分解为正弦函数的和。其他的过程都和上述过程大致相同，不同的是在输出端，经过调整幅度后的正弦函数之和产生奇的输出信号 $h(t)$。

类似地，如果 $f(t)$ 是一个非奇非偶的函数，那么它可以先被分解成奇函数和偶函数，然后再像上面的步骤进行分解，将其分解成正弦和余弦函数。这些正弦和余弦函数在输出端的和产生输出信号 $h(t)$。

在上面的讨论中，使用的传递函数是一个实的偶函数。现在假设输入的是一个实偶函数，而冲激响应是一个实奇函数，这将使传递函数成为虚奇函数。当输入的偶脉冲对乘以虚奇传递函数后，被转化成虚的奇冲激对。这个过程将输入端的余弦函数转化为输出端的正弦函数，输出变为正弦函数的和，而 $g(t)$ 是奇函数。

由此知道，一个偶的输入函数同一个奇的冲激响应做卷积，会产生一个奇函数。从卷积的图示过程中可以看出这个是正确的。

现在输入函数为余弦函数，即：

$$f(t)=\cos(2\pi f_0 t) \qquad f_0 \geqslant 0 \tag{4.3.20}$$

冲激响应是实函数，它包括奇部和偶部：

$$g(t)=g_e(t)+g_0(t) \tag{4.3.21}$$

通过傅立叶变换得到它的传递函数：

$$G(s)=G_e(s)+jG_0(s) \tag{4.3.22}$$

是 Hermite 函数，即：

$$G(f_0)=G_e(f_0)+jG_0(f_0) \qquad f_0 \geqslant 0 \tag{4.3.23}$$

且：

$$G(-f_0)=G^*(f_0)=G_e(f_0)-jG_0(f_0) \qquad f_0 \geqslant 0 \tag{4.3.24}$$

余弦函数的谱可以写作：

$$F(s)=\frac{1}{2}\left[\delta(s-f_0)+\delta(s+f_0)\right] \tag{4.3.25}$$

这样可以将输出谱写成：

$$H(s)=\frac{1}{2}G_e(f_0)\left[\delta(s-f_0)+\delta(s+f_0)\right]+j\frac{1}{2}G_0(f_0)\left[\delta(s-f_0)-\delta(s+f_0)\right] \tag{4.3.26}$$

这说明输出信号为：

$$h(t)=G_e(f_0)\cos(2\pi f_0 t)+G_0(f_0)\sin(2\pi f_0 t) \tag{4.3.27}$$

也可以写作：

$$h(t)=A\cos(2\pi f_0 t+\phi) \tag{4.3.28}$$

其中：

$$A=\sqrt{G_e^2(f_0)+G_0^2(f_0)} \quad 和 \quad \phi=\arctan\left[\frac{G_0(f_0)}{G_e(f_0)}\right] \tag{4.3.29}$$

这正是分析线性系统时的预期结果，即它可以改变正弦型输入的幅值和相位，但是不改变其频率和函数形式。上面的推导阐述了冲激响应的奇部和偶部与传递函数的实部和虚部之间的关系：

（1）冲激响应的奇部是在传递函数中引入一项虚奇部的，因此会使得输入的余弦产生一个正弦型的输出并且在输出端发生相移。

（2）它表明输出的幅值取决于复传递函数的幅值（模）的均方根。

至此有了分析线性系统的两种等价的方法：

（1）可以将卷积形象化，即将函数反折、位移、相乘、积分。

（2）可以将正弦函数分解形象化，然后相乘并再求和。

一个域中函数的奇偶性会对另一个域中的函数的奇偶性产生约束。有了上述两种方法，人们在用线性系统分析方法来解决一个问题时就有了很大的灵活性，就如同是两种语言的互补。

4.3.4 复频率

波形和频谱分析器中有窄带通滤波器，它们只允许某一个特定的小的（正弦）频率范围内的能量通过。这些滤波器被用来从信号中提取出某一特定的正弦部分。

可以通过窄（正）频率范围内调窄带通滤波器，同时测出输出的幅值，就能够得到电信号的谱。

余弦函数的傅立叶变换是一个偶冲激对，正弦函数的傅立叶变换是一个虚奇冲激对。由于余弦函数是偶函数，因此它的谱必定是偶的，正弦函数则类似地有奇函数的特性。

对于任意的一个实函数，它的谱的左一半是冗余的，因此可以被忽略。正如窄谱分析器中就忽略了这一项。但是在使用更一般的数学方法来建立物理过程的模型，保留函数的左边一半在数学上会使得分析简单得多。这里都画出了双边的谱，虽然在其他地方常常只画出正频率的谱。

注意：只要在建立线性系统的模型时采用的双边数学，函数的左边一半则是冗余的，但是是分析的一部分。

4.4 二维傅立叶变换

到目前为止，讨论的都是一维时域函数的傅立叶变换。但是在数字图像处理中，或是在光系统的分析过程中，输入和输出通常都是二维的，有时是高维的。有了前面一维的基础，理解高维的形式才更容易。

4.4.1 二维傅立叶变换的定义

图像是二维的，所以需要二维的傅立叶变换，对于一幅图像 $f(x,y)$，二维的傅立叶变换为：

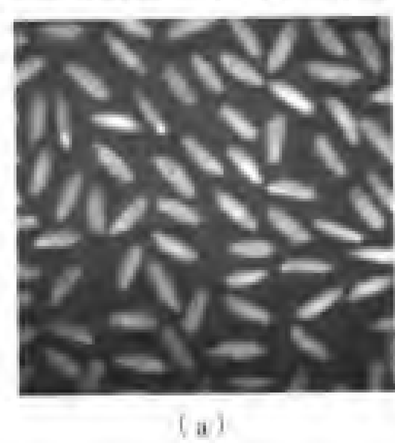
$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy \tag{4.4.1}$$

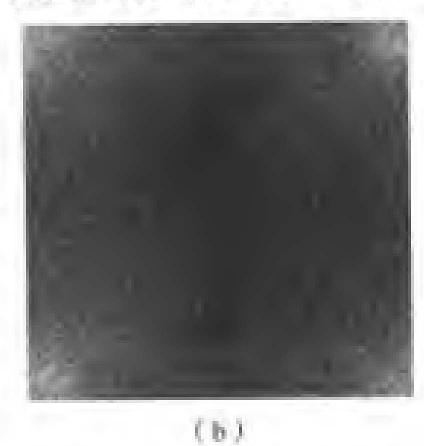
反变换为：

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(u,v)e^{j2\pi(ux+vy)}dudv \tag{4.4.2}$$

其中，$F(u,v)$ 是谱函数有两个实频率变量 u 和 v，u 对应 x 轴，v 对应 y 轴。

例：如图 4-8 所示是一幅图像（a）及其二维的幅度谱（b），（c）是将（b）中的直流分量移到频谱的中心，常用于傅立叶变换的可视化。二维频率空间（b）的每个点的幅值（实部和虚部的平方和的平方根）被规格化为显示灰度级。如图 4-8（b）所示。


（a）


（b）

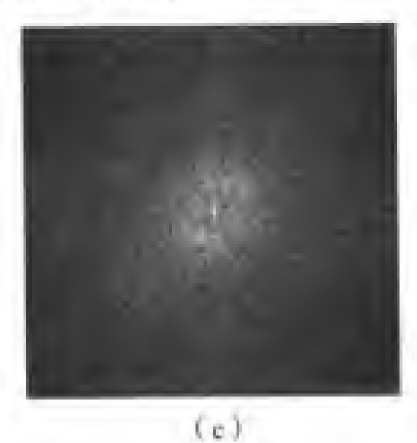

（c）

图 4-8 二维傅立叶变换

（a）一幅 rice 的数字图像。

（b）rice 图像的二维傅立叶变换。

（c）通过将直流分量移到图像中间，可视化处理。

图 4-8（b）中图像四个角的白色部分是图像经过变换得到的谱的高频部分，而（c）中将高频部分移到图像中间，便于可视化。

4.4.2 二维离散傅立叶变换（DFT）

如果 $g(i,k)$ 是一个 $N\times N$ 的（就好像用等间距的矩形网格对一个二维连续函数采样所得的）数组，则它的二维傅立叶变换为：

$$G(m,n)=\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}g(i,k)e^{-j2\pi\left(m\frac{i}{N}+n\frac{k}{N}\right)} \tag{4.4.3}$$

反变换（逆 DFT）为：

$$g(i,k)=\frac{1}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}G(m,n)e^{j2\pi\left(\frac{m}{N}i+k\frac{n}{N}\right)} \tag{4.4.4}$$

和一维 DFT 的情况一样，DFT 和连续傅立叶变换很相似。一个在矩形网格上采样的带宽有限函数的二维 DFT 是连续傅立叶变换的一个特例。由于它们的相似性，所以在后面二维的性质的讨论中可以看见，它们具有几乎一样的性质。

由于式（4.4.3）中的指数可以被分解，从而变换可分离为：

$$G(m,n)=\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}\left[\frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}g(i,k)e^{-j2\pi\left(n\frac{k}{N}\right)}\right]e^{-j2\pi\left(m\frac{i}{N}\right)} \tag{4.4.5}$$

由此将 DFT 分解为水平和垂直两部分运算。在式（4.4.5）方括号中的项表示在图像上的行进行 DFT 计算，方括号外面的求和则实现结果数组在列上进行 DFT 计算。这种分解

可以使人们能用一维快速傅立叶变换（FFT）来快速实现二维 DFT。二维逆 DFT 也同样可分离，如图 4-9 所示。



图 4-9 二维傅立叶变换的可分离性

4.4.3 二维傅立叶变换的矩阵表示

DFT 用矩阵符号表示为：

$$G = FgF \tag{4.4.6}$$

其中：

$$F = [f_{ik}] = \left[\frac{1}{\sqrt{N}}\right] e^{-j2\pi ik/N} \tag{4.4.7}$$

是一个 $N \times N$ 的复系数矩阵。

可以证明 F 是一个酉矩阵，即矩阵的逆是它的复共轭的转置（$F^{-1} = (F^*)^T$）。要得到一个酉矩阵的逆，只需要简单地交换行和列的位置，并改变每个元素的虚部的符号。由于 F 还是对称的，所以转置又可以省略掉了，即：$F^{-1} = F^*$。

注意：计算二维卷积时，通常将各行堆积为一个大的列向量，并用大的循环矩阵，但在计算二维 DFT 时不必这样。这是因为 DFT 变换核函数可被分解为行和列的一维运算，而且 F 是一个酉矩阵。

4.4.4 二维傅立叶变换的性质

（1）一维傅立叶变换的性质有：对称性、加法定理、位移定理、卷积定理、相似性定理、Rayleigh 定理。

（2）二维傅立叶变换特有的性质：旋转、可分离乘积，拉普拉斯。

表 4-3 归纳了二维傅立叶变换的定理。从一维到二维的推广是很直接的。二维傅立叶变换特有的性质中，可分离乘积在上一节已经介绍过了，一幅二维图像可以分解为一维分量的乘积，对于二维图像的谱也一样。另一个是旋转性质，这在计算机断层造影技术（CAT）中很有用。

表 4-3 二维傅立叶变换的性质

性质	空间域	频域
加法定理	$f(x,y)+g(x,y)$	$F(u,v)+G(u,v)$
相似性定理	$f(ax,by)$	$\frac{1}{\lvert ab\rvert}F\left(\frac{u}{a},\frac{v}{b}\right)$
位移定理	$f(x-a,y-b)$	$e^{-j2\pi(au+bv)}F(u,v)$
卷积定理	$f(x,y)*g(x,y)$	$F(u,v)G(u,v)$
旋转	$f(x\cos\theta+y\sin\theta,-x\cos\theta+y\sin\theta)$	$F(u\cos\theta+v\sin\theta,-u\cos\theta+v\sin\theta)$
Rayleigh 定理	$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\lvert f(x,y)\rvert^2 dxdy$	$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\lvert F(u,v)\rvert^2 dudv$

续表 4-3

性质	空间域	频域
微分	$\left(\frac{\partial}{\partial x}\right)^m\left(\frac{\partial}{\partial y}\right)^n f(x,y)$	$(j2\pi u)^m(j2\pi v)^n F(u,v)$
可分离定理	$f(x)g(y)$	$F(u)G(v)$
拉普拉斯	$\nabla^2 f(x,y)=\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}f(x,y)$	$-4\pi^2(u^2+v^2)F(u,v)$

再有一个特别的性质就是有关拉普拉斯算子的。拉普拉斯是一个全向的二阶微分算子，通常用作边缘检测和边缘增强。由卷积定理，拉普拉斯算子对应的线性系统特征为：传递函数随频率的平方增加。

1. 可分离性

假设输入二维函数 $f(x,y)$ 可以分解成两个一维函数的乘积：

$$f(x,y)=f_1(x)f_2(y) \tag{4.4.8}$$

则有：

$$F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f_1(x)f_2(y)e^{-j2\pi(ux+vy)}dxdy \tag{4.4.9}$$

整理得：

$$F(u,v)=\int_{-\infty}^{\infty}f_1(x)e^{-j2\pi ux}dx\int_{-\infty}^{\infty}f_2(y)e^{-j2\pi vy}dy=F_1(u)F_2(v) \tag{4.4.10}$$

由此可以看到，当一个二维图像分解为两个一维分量，它的频谱也被分解为两个一维分量。

例如椭圆高斯函数：

$$e^{-(x^2/2\delta_x^2+y^2/2\delta_y^2)}=e^{-x^2/2\delta_x^2}e^{-y^2/2\delta_y^2} \tag{4.4.11}$$

它可以被分解为两个一维高斯函数的乘积，若标准方差 $\delta_x=\delta_y$，则上式变为：

$$e^{-(x^2+y^2)/2\delta^2}=e^{-x^2/2\delta^2}e^{-y^2/2\delta^2} \tag{4.4.12}$$

它是一个圆高斯函数。这个函数在光学分析中十分重要，由于它具有圆对称性，可以被分解为一维函数的乘积。

2. 相似性

相似性定理推广到二维是很直接的，可以得到：

$$F\{f(a_1x+b_1y,a_2x+b_2y)\}$$
$$=(A_1B_2-A_2B_1)F(A_1u+A_2v,B_1u+B_2v) \tag{4.4.13}$$

其中：

$$F\{f(x,y)\}=F(u,v) \tag{4.4.14}$$

$$A_1=\frac{b_2}{a_1b_2-a_2b_1} \qquad A_2=\frac{-a_2}{a_1b_2-a_2b_1}$$
$$B_1=\frac{-b_1}{a_1b_2-a_2b_1} \qquad B_2=\frac{a_1}{a_1b_2-a_2b_1} \tag{4.4.15}$$

令：

$$w=a_1x+b_1y,\qquad z=a_2x+b_2y \tag{4.4.16}$$

从而：

$$x = A_1 w + B_1 z, \quad y = A_2 w + B_2 z \qquad (4.4.17)$$

J 行列式为：

$$J = \begin{vmatrix} \frac{\partial x}{\partial w} & \frac{\partial y}{\partial w} \\ \frac{\partial x}{\partial z} & \frac{\partial y}{\partial z} \end{vmatrix} = \begin{vmatrix} A_1 & A_2 \\ B_1 & B_2 \end{vmatrix} = A_1 B_2 - A_2 B_1 \qquad (4.4.18)$$

于是：

$$\begin{aligned} & F\{f(a_1 x + b_1 y, a_2 x + b_2 y)\} \\ & = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(w,z) e^{-j2\pi(A_1 u + A_2 v)w + (B_1 u + B_2 v)z} \begin{vmatrix} \frac{\partial x}{\partial w} & \frac{\partial y}{\partial w} \\ \frac{\partial x}{\partial z} & \frac{\partial y}{\partial z} \end{vmatrix} dzdw \\ & = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(w,z) e^{-j2\pi(A_1 u + A_2 v)w + (B_1 u + B_2 v)z} dzdw (A_1 B_2 - A_2 B_1) \\ & = (A_1 B_2 - A_2 B_1) F(A_1 u + A_2 v, B_1 u + B_2 v) \end{aligned} \qquad (4.4.19)$$

3. 旋转

由二维的相似性定理可知，如果在空间域中函数 $f(x,y)$ 旋转一个角度，则 $f(x,y)$ 的谱函数也旋转相同的角度。

设 $f(x,y)$ 旋转了 θ，则：

$$\begin{aligned} & F\{f(x\cos\theta + y\sin\theta, -x\sin\theta + y\cos\theta)\} \\ & = F(u\cos\theta + v\sin\theta, -u\sin\theta + v\cos\theta) \end{aligned} \qquad (4.4.20)$$

证明：

$f(x,y)$ 旋转角度 θ 的表达为：

$$f(\cos\theta x + \sin\theta y, -\sin\theta x + \cos\theta y) \qquad (4.4.21)$$

比较（4.4.13）式，令：

$$a_1 = \cos\theta, b_1 = \sin\theta \qquad a_2 = -\sin\theta, b_2 = \cos\theta \qquad (4.4.22)$$

则：

$$A_1 = \cos\theta, A_2 = \sin\theta \qquad B_1 = -\sin\theta, B_2 = \cos\theta \qquad (4.4.23)$$

于是：

$$\begin{aligned} & F\{f(x\cos\theta + y\sin\theta, -x\sin\theta + y\sin\theta)\} \\ & = F(u\cos\theta + v\sin\theta, -u\sin\theta + v\cos\theta) \end{aligned} \qquad (4.4.24)$$

即 $f(x,y)$ 旋转角度 θ，$f(x,y)$ 的谱也旋转角度 θ。

4. 投影

将二维函数 $f(x,y)$ 投影到 x 轴上，得到 x 轴上的一个一维函数：

$$p(x) = \int_{-\infty}^{\infty} f(x,y)dy \qquad (4.4.25)$$

则 $p(x)$ 的傅立叶变换为：

$$P(u) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)dy \cdot e^{-j2\pi ux} dx \qquad (4.4.26)$$

于是 $P(u)$ 可以写为：

$$P(u) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y) e^{-j2\pi(ux - 0y)} dxdy = F(u,0) \qquad (4.4.27)$$

$f(x,y)$在x轴上投影的傅立叶变换，即为$F(u,v)$在u轴上取值（如图 4-10 所示）。

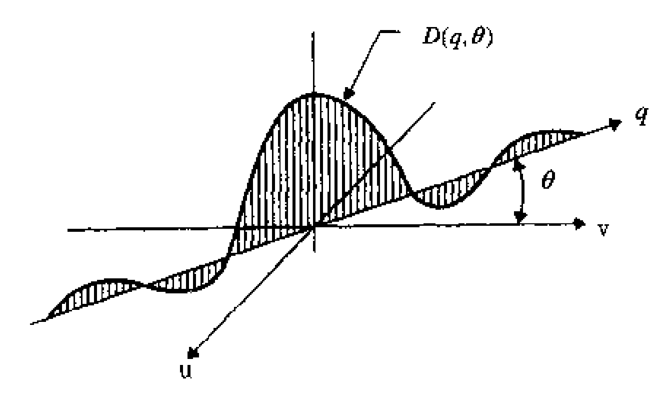


图 4-10 二维傅立叶变换的投影性质

5. 一些其他的性质

（1）位移性。

$$f(x-x_0,y-y_0) \Leftrightarrow F(u,v)\exp(-j2\pi\frac{x_0u+y_0v}{N})$$

$$f(x,y)\exp(j2\pi\frac{xu_0+yv_0}{N}) \Leftrightarrow F(u-u_0,v-v_0) \tag{4.4.28}$$

（2）周期性。

$$F(u,v)=F(u+N,v)=F(u,v+N)=F(u+N,v+N) \tag{4.4.29}$$

（3）共轭对称性。

$$F(u,v)=F^*(-u,-v) \tag{4.4.30}$$

（4）SHIFT：通常将变换的原点移到$(\frac{N}{2},\frac{N}{2})$处，以看到完全的频谱，如图 4-8 所示。

$$f(x,y)(-1)^{x+y} \Leftrightarrow F(u-\frac{N}{2},v-\frac{N}{2}) \tag{4.4.31}$$

① 若$f(x,y)$在与x轴成θ角的直线上投影，由旋转定理，其傅立叶变换正好等于$F(u,v)$沿与u轴成θ角的直线上的取值。

② 二维傅立叶变换的性质，见表 4-3。

特别的是：拉普拉斯算子是一个全向二阶微分算子，通常作边缘增强和边缘检测。对函数使用拉普拉斯后作变换，相当于在它的谱乘上u^2+v^2。

4.4.5 圆对称和 Hankel 变换

1. 圆对称

许多重要的二维函数都具有圆对称性，这样就可以引入极坐标的概念，将一个二维函数表示为单个自变量（半径）的函数：

$$f(x,y)=f_r(r) \tag{4.4.32}$$

其中r表示半径：

$$r^2=x^2+y^2 \tag{4.4.33}$$

现在可以将$f(x,y)$的傅立叶变换写成极坐标的形式：

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(x,y)e^{-j2\pi(ux+vy)}dxdy=\int_0^{\infty}\int_0^{2\pi}f_r(r)e^{-j2\pi qr\cos(\theta-\phi)}rdrd\theta \tag{4.4.34}$$

这里已经将积分从直角坐标系变换到了极坐标系，其中变量替换如下：

$$x + jy = re^{j\theta} \qquad u + jv = qe^{j\phi} \tag{4.4.35}$$

由于积分区间是余弦函数的整个周期，将式（4.4.34）中的 ϕ 去掉：

$$F\{f(x,y)\} = \int_0^{\infty} f_r(r)\left[\int_0^{2\pi} e^{-j2\pi qr\cos(\theta)} d\theta\right] rdr \tag{4.4.36}$$

现在来看方括号中的积分，第一类零阶的 Bessel 函数的定义为：

$$J_0(z) = \frac{1}{2\pi}\int_0^{2\pi} e^{-jz\cos(\theta)} d\theta \tag{4.4.37}$$

因此将式（4.4.32）写成：

$$F\{f(x,y)\} = 2\pi\int_0^{\infty} f_r(r) J_0(2\pi qr) rdr \tag{4.4.38}$$

注意：圆对称函数的傅立叶变换是个单自变量函数，自变量为径向频率 q 的函数。

其中：

$$q = u^2 + v^2 \tag{4.4.39}$$

2. Hankel 变换

定义 Hankel 变换，对于圆对称函数，正变换为：

$$F_r(q) = 2\pi\int_0^{\infty} f_r(r) J_0(2\pi qr) rdr \tag{4.4.40}$$

反变换为：

$$f_r(r) = 2\pi\int_0^{\infty} F_r(q) J_0(2\pi qr) qdq \tag{4.4.41}$$

上述两式定义了一阶 Hankel 变换，其实它是二维傅立叶变换的一个特例。这种变换是一维的线性积分变换，它和傅立叶变换类似，只是积分核是 Bessel 函数 $J_l(z)$（I 代表 Bessel 函数的阶数）。

表 4-4 是某些函数的 Hankel 变换，表 4-5 列出了 Hankel 变换的一些性质。

表 4-4　某些函数的 Hankel 变换

函数	$f(r)$	$E(q)$
倒数	$\frac{1}{r}$	$\frac{1}{q}$
高斯函数	$e^{-\pi r^2}$	$e^{-\pi q^2}$
冲激函数	$\frac{\delta(r)}{\pi r}$	1
矩形脉冲	$\Pi\left(\frac{r}{2a}\right)$	$\frac{aJ_1(2\pi aq)}{q}$
三角脉冲	$\Lambda\left(\frac{r}{2a}\right)$	$\frac{2\pi}{ax^3}\int_0^x J_0(x)dx - \frac{2\pi}{ax^2}J_0(x)$
位移冲激	$\delta(r-a)$	$2\pi aJ_0(2\pi aq)$
指数延迟	e^{-ar}	$\frac{2\pi a}{\left[(2\pi q)^2 + a^2\right]^{3/2}}$

续表 4-4

函数	$f(r)$	$E(q)$
指数延迟	$\dfrac{e^{-ar}}{r}$	$\dfrac{2\pi}{\left[(2\pi q)^2+a^2\right]^{1/2}}$
	$\pi r^2 e^{-\pi r^2}$	$\left(\dfrac{1}{\pi}-q^2\right)e^{-\pi q^2}$
	$\dfrac{\sin(2\pi ar)}{r}$	$\dfrac{\Pi(q/2a)}{\sqrt{a^2-q^2}}$

表 4-5 Hankel 变换的性质

性质	空间域	频域
加法定理	$f(r)+g(r)$	$F(q)+G(q)$
相似性定理	$f(ar)$	$\dfrac{1}{a^2}F\left(\dfrac{q}{a}\right)$
卷积定理	$\int_0^{\infty}\int_0^{2\pi} f(\rho)g(r^2+\rho^2-2r\rho\cos\theta)\rho d\rho d\theta$	$F(q)G(q)$
拉普拉斯	$\nabla^2 f(r)=\dfrac{d^2 f}{dr^2}+\dfrac{1}{r}\dfrac{df}{dr}$	$-4\pi^2 q^2 F(q)$
Rayleigh 定理	$\int_0^{\infty}\lvert f(r)\rvert^2 rdr=E$	$\int_0^{\infty}\lvert f(q)\rvert^2 qdq=E$
能量定理	$\int_0^{\infty} f(r)g^*(r)rdr=P$	$\int_0^{\infty} F(q)G^*(q)qdq=P$

4.5 相关性和能量谱概念

4.5.1 互相关

给定两个函数 $f(t)$ 和 $g(t)$，它们的互相关函数定义为：

$$R_{fg}(\tau)=f(t)*g(-t)=\int_{-\infty}^{\infty} f(t)g(t+\tau)dt \tag{4.5.1}$$

$R_{fg}(\tau)$ 反映了两个函数在不同的相对位置上互相匹配的程度。

互相关函数的傅立叶变换称为互能量谱或互能量谱密度函数：

$$P_{fg}(s)=F\left\{\hat{R}_{fg}(\tau)\right\}=F(s)G^*(s) \tag{4.5.2}$$

4.5.2 自相关

对应于互相关函数，将式（4.5.1）中第一个等号后面的 $g(-t)$ 用 $f(-t)$ 代替，就得到了自相关函数的定义如下：

$$R_f(\tau)=f(t)*f(-t)=\int_{-\infty}^{\infty} f(t)f(t+\tau)dt \tag{4.5.3}$$

性质：

（1）自相关函数 $R_f(\tau)$ 永远是偶函数：

$$R_f(-\tau)=\int_{-\infty}^{\infty} f(t)f(t-\tau)dt=\int_{-\infty}^{\infty} f(t+\tau)f(t)dt=R_f(\tau) \tag{4.5.4}$$

（2）在 $\tau=0$ 时 $R_f(\tau)$ 达到极大值。

（3）存在等式：

$$\int_{-\infty}^{\infty} R_f(\tau)d\tau = \left[\int_{-\infty}^{\infty} f(t)dt\right]^2 \tag{4.5.5}$$

证明：

$$\begin{aligned}\int_{-\infty}^{\infty} R_f(\tau)d\tau &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(t)f(t+\tau)dxd\tau = \int_{-\infty}^{\infty} f(t)\left[\int_{-\infty}^{\infty} f(\tau+t)d\tau\right]dt \\ &= \int_{-\infty}^{\infty} f(s)ds\int_{-\infty}^{\infty} f(t)dt\end{aligned} \tag{4.5.6}$$

（4）每个函数有惟一自相关函数，反之不然。

4.5.3　能量谱（功率谱）

自相关的傅立叶变换为：

$$\begin{aligned}P_f(s) &= F\left\{R_f(\tau)\right\} = F\left\{f(t)*f(-t)\right\} \\ &= F(s)F(-s) = F(s)F^*(s) = |F(s)|^2\end{aligned} \tag{4.5.7}$$

或者：

$$\begin{aligned}P_f(s) = \int_{-\infty}^{\infty} R_f(\tau)e^{-j2\pi s\tau}d\tau &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(t)f(\tau+t)e^{-j2\pi s\tau}dtd\tau \\ &= \int_{-\infty}^{\infty} f(t)e^{+j2\pi st}dt\int_{-\infty}^{\infty} f(\tau+t)e^{-j2\pi s(\tau+t)}d\tau \\ &= F(s)F(-s)\end{aligned} \tag{4.5.8}$$

（1）$P_f(s)$ 称为 $f(t)$ 的能量密度函数，能量谱（功率谱）。

（2）若 $f(t)$ 为实数，则 $R_f(t)$ 为实偶，于是 $P_f(s)$ 也是实偶的。

（3）$f(t)$ 有惟一能量谱，反之不然。

4.6　傅立叶变换性质总结

本章介绍了傅立叶变换的许多有用的性质，在以后的图像处理系统分析中将常用到它们。为了查阅方便，表 4-6 列出了这些性质。

表 4-6　傅立叶变换性质总结

性质	时间（或空间）域	频域
术语	信号 冲激响应 自相关函数 交叉相关函数 $f(x)$	谱 传递函数 能量谱 交叉能量谱 $F(s)$
定义	$\int_{-\infty}^{\infty} F(s)e^{j2\pi sx}ds$	$\int_{-\infty}^{\infty} f(x)e^{-j2\pi sx}dx$
加法定理	$af(x)+bg(x)$	$aF(s)+bG(s)$
相似性定理	$f(ax)$	$\frac{1}{\lvert a\rvert}F\left(\frac{s}{a}\right)$
位移定理	$f(x-a)$	$e^{-j2\pi as}F(s)$
卷积定理	$f(x)*g(x)$	$F(q)G(q)$

续表 4-6

性质	时间（或空间）域	频域
微分	$\frac{d}{dx}f(x)$	$j2\pi sF(s)$
自相关定理	$R_f(\tau)=f(x)*f^*(-x)$	$\|F(s)\|^2=P_f(s)$
Rayleigh 定理	$\int_{-\infty}^{\infty}\|f(x)\|^2 xdx=E$	$\int_{-\infty}^{\infty}\|F(s)\|^2 sds=E$
功率定理	$\int_{-\infty}^{\infty}f(x)g^*(x)xdx=P$	$\int_{-\infty}^{\infty}F(s)G^*(s)sds=P$

小结

本章主要介绍了傅立叶变换的概念和性质、线性系统和傅立叶变换、二维傅立叶变换、相关性和能量谱概念、傅立叶变换性质总结等内容。

重点： 本章的重点主要有以下几个方面：

（1）傅立叶变换是线性积分变换，它在时间（或空间）域的复函数和频率域的复函数之间建立起惟一的对应。

（2）高斯函数的傅立叶变换是另一个高斯函数。

（3）傅立叶变换保持奇偶性。

（4）实函数的傅立叶变换是 Hermite 函数。

（5）位移函数的原点将在谱中引入一个相位移（与频率成正比），从而改变了谱的实部和虚部的能量分配，但不改变总能量（位移定理）。

（6）函数和的傅立叶变换等于它们分别做变换再求和（加法定理）。

（7）两个函数的卷积对应于它们的傅立叶变换的相乘，同样在频域中做卷积对应于时域（或空间域）中做乘积。

（8）压缩一个函数会扩展它的傅立叶变换，反之亦然（相似性定理）。

（9）函数（信号）的能量通其傅立叶变换谱的能量相等（能量守恒）。

（10）线性系统的传递函数可以由输出谱（可测量）和输入谱（已知）的比值得到。

（11）正弦型函数的傅立叶变换是一个等间隔冲激对。

（12）输入信号可被分解为无限多个无限小的正弦型函数的和。

（13）线性系统可以想象成对输入信号的各个正弦型分量单独作用，然后在输出端求和形成输出信号。

（14）傅立叶变换推广到二维或更高维是很直接的。

（15）如果一个双变量函数可以被分解为两个单变量函数的乘积，则它的傅立叶变换也可以被分解（可分离定理）。

（16）旋转一个二维函数，它的傅立叶变换也旋转相同角度（旋转定理）。

（17）将一个二维函数投影到一条与 X 轴成 θ 角的直线上，再对得到的一维函数作傅立叶变换，结果为二维谱在与 u 轴成 θ 的直线上的取值（投影定理）。

（18）圆对称二维函数具有圆对称性的谱。

（19）Hankel 变换建立了圆对称函数的剖面函数及其谱的关系。

（20）互相关和卷积类似，不过在过程中并不反折任一函数。

（21）自相关和互相关类似，是不反折任一函数的自卷积。

（22）自相关函数的傅立叶变换是能量谱。

难点：如何理解时域与频域的关系，以及快速傅立叶变换的原理。

疑点：傅立叶变换的缺陷在哪里。

练习四

一、选择题

1. 冲激函数 $\delta(t)$ 的傅立叶变换为（　　）。

A. 1　　B. 2　　C. 0.1　　D. $\frac{\sin(\pi s)}{\pi s}$

2. 矩形脉冲 $\Pi(t)$ 的傅立叶变换为（　　）。

A. 1　　B. $e^{-\pi s^2}$　　C. $\frac{\sin^2(\pi s)}{(\pi s)^2}$　　D. $\frac{\sin(\pi s)}{\pi s}$

3. 余弦函数 $\cos(2\pi f t)$ 的傅立叶变换为（　　）。

A. $\frac{1}{2}\left[\delta(s)-\frac{j}{\pi s}\right]$　　B. $\delta(s-f)$

C. $\frac{1}{2}[\delta(s+f)+\delta(s-f)]$　　D. $\frac{\sin(\pi s)}{\pi s}$

二、思考题

1. 在对线性系统分析的研究中，局限于讨论在这个已发展成熟的领域中的某一部分，例如只用傅立叶变换而不用拉普拉斯变换，或者是 Z 变换，原因是什么？

2. 一幅图像，经过傅立叶变换之后，将高频部分删除，在进行反变换，设想一下将会得到什么结果？

三、上机题

1. z 为一个 3*3 的复矩阵，计算它的实部与虚部、幅值与幅角。

2. 编写一个程序，将一幅图像进行傅立叶变换，然后再进行傅立叶反变换，观察得到的图像和原始图像有无差别。

3. 编写一个程序，将一幅图像进行二维傅立叶变换和反变换，并将 0 频率分量移到矩阵的中心。

第 5 章　采样数据处理

在数字图像处理中，通过采样得到输入，然后通过插值复原。在整个过程中，如果采样得当，并且进行了正确的处理，不会使从连续函数分析中获得的结果失效。因此本章将对采样过程进行简明而有效的分析。主要研究连续图像采样和采样数据处理的各个方面。

本章主要内容如下：

（1）预备知识。

（2）采样和插值。

（3）频谱计算。

（4）混叠。

（5）截取。

（6）数字处理简介。

（7）混叠误差的控制。

5.1　预备知识

回顾 $\delta(x)$ 的性质：

（1）$\int_{-\infty}^{\infty} f(x+x_0)\delta(x)dx = f(x_0)$

（2）$\int_{-\infty}^{\infty} f(x)\delta(x-x_0)dx = \int_{-\infty}^{\infty} f(x+x_0)\delta(x)dx = f(x_0)$

（3）$\delta(ax) = \frac{1}{|a|}\delta(x)$

（4）$\delta(x) * f(x) = f(x)$

假设：

$$III(x) = \sum_{n=-\infty}^{\infty} \delta(x-n) \tag{5.1.1}$$

即 III(x)是一个沿 x 轴相隔单位间距出现的单位幅值冲激序列。

试证明：

$$F\{III(x)\} = III(s) \tag{5.1.2}$$

其中 F 是对其做傅立叶变换。

证明：

由于 $III(x)$ 是以 1 为周期的函数，按傅立叶级数展开，有：

$$III(x) = \sum_{m=-\infty}^{\infty} C_m e^{j2\pi mx} \tag{5.1.3}$$

可以证明：

$$C_m = \int_{-\frac{1}{2}}^{\frac{1}{2}} \sum_{k=-\infty}^{\infty} \delta(x-k) e^{-j2\pi mx} dx = 1 \tag{5.1.4}$$

即：

$$
\begin{aligned}
C_m &= \int_{-\frac{1}{2}}^{\frac{1}{2}} \sum_{k=-\infty}^{\infty} \delta(x-k) e^{-i2\pi m x} dx \\
&= \sum_{k=-\infty}^{\infty} \int_{-\frac{1}{2}}^{\frac{1}{2}} \delta(x-k) e^{-i2\pi m x} dx \\
&= \sum_{k=-\infty}^{\infty} \int_{-\frac{1}{2}-k}^{\frac{1}{2}-k} \delta(x) e^{-i2\pi m(x+k)} dx \\
&= \sum_{k=-\infty}^{\infty} \int_{-\frac{1}{2}-k}^{\frac{1}{2}-k} \delta(x) e^{-i2\pi m x} dx \cdot e^{-i2\pi m k}
\end{aligned} \tag{5.1.5}
$$

由于：

$$
e^{-i2\pi k m} = 1 \tag{5.1.6}
$$

所以：

$$
\begin{aligned}
C_m &= \sum_{k=-\infty}^{\infty} \int_{-\frac{1}{2}-k}^{\frac{1}{2}-k} \delta(x) e^{-i2\pi m x} dx \\
&= \int_{-\infty}^{\infty} \delta(x) e^{-i2\pi m x} dx \\
&= 1
\end{aligned} \tag{5.1.7}
$$

于是 $III(x)$ 的傅立叶变换为：

$$
\begin{aligned}
F\{III(x)\} &= \int_{x=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} C_m e^{j2\pi m x} e^{-j2\pi s x} dx \\
&= \sum_{m=-\infty}^{\infty} \int_{x=-\infty}^{\infty} e^{j2\pi m x} e^{-j2\pi s x} dx \\
&= \sum_{m=-\infty}^{\infty} \delta(s-m)
\end{aligned} \tag{5.1.8}
$$

因为：

$$
\int_{x=-\infty}^{\infty} e^{i2\pi m x} e^{-j2\pi s x} dx = \delta(s-m) \tag{5.1.9}
$$

Shah 函数：

Shah 函数，也称梳状函数，是一个无限冲激串（序列）。它应读作“Shah of x”，其定义如下：

$$
III(x) = \sum_{n=-\infty}^{\infty} \delta(x-n) \tag{5.1.10}
$$

$III(x)$ 是一个沿 x 轴相隔单位间距出现的单位幅值冲激序列，Shah 函数的傅立叶变换是其本身，即：

$$F\{III(x)\}=III(s) \tag{5.1.11}$$

由相似性定理：

$$F\{f(at)\}=\frac{1}{|a|}F(\frac{s}{a}) \tag{5.1.12}$$

所以有：

$$F\{III(\frac{x}{\tau})\}=\tau\, III(\tau s) \tag{5.1.13}$$

即其频谱为沿 s 轴间隔为 $1/\tau$ 的冲激序列（如图 5-1 所示）。

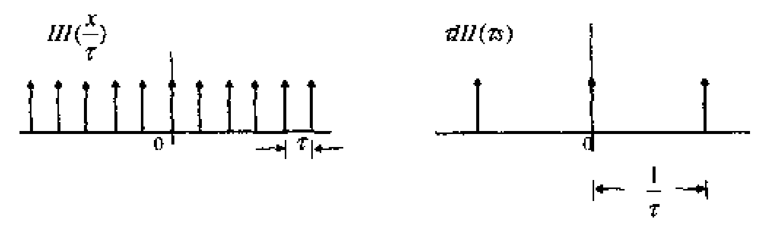


图 5-1 Shah 函数及其频谱

由预备知识知：

$$\delta(ax)=\frac{1}{|a|}\delta(x) \tag{5.1.14}$$

则强度：

$$III(\frac{x}{\tau})=\sum_{n=-\infty}^{\infty}\delta(\frac{x}{\tau}-n)=\tau\sum_{n=-\infty}^{\infty}\delta(x-n\tau) \tag{5.1.15}$$

冲激间隔为 τ 而不是单位间距，它使得冲激强度的幅值乘了系数 τ。

而 $III(\frac{x}{\tau})$ 的傅立叶变换：

$$F\{III(\frac{x}{\tau})\}=\tau\, III(\tau s)=\tau\cdot\frac{1}{|\tau|}\sum_{n=-\infty}^{\infty}\delta(s-\frac{n}{\tau})=\sum_{n=-\infty}^{\infty}\delta(s-\frac{n}{\tau}) \tag{5.1.16}$$

即：在时域中一个强度为 τ，间隔为 τ 的冲激序列，在频域中产生一个间隔为 $1/\tau$ 的单位冲激序列。

5.2 采样和插值

5.2.1 使用 Shah 函数采样及其频谱

对于有限带宽的函数 f(x)，设带宽为 s_0，即：

$$F(s)=0 \qquad |s|\geqslant s_0 \tag{5.2.1}$$

如图 5-2 所示。

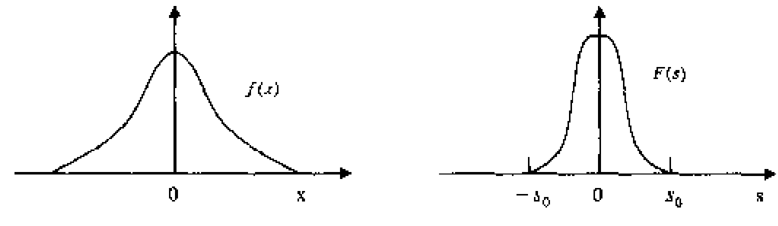


图 5-2 频带受限制的函数

如果以等间隔τ对时域中的 $f(x)$ 采样，则仅在 $x = n\tau$ 处取 $f(x)$ 的值，在其他的地方 $f(x)$ 被破坏了。将采样过程模型化为简单地用 $\text{III}(x/\tau)$ 乘以函数 $f(x)$ 而得到采样后的函数 $g(x)$。这个过程中，各采样点之间的函数值为零，而在采样点上由于冲激函数的性质，而保存了函数值。采样后的函数如图 5-3 所示，左图为时域函数，右图为频域函数。

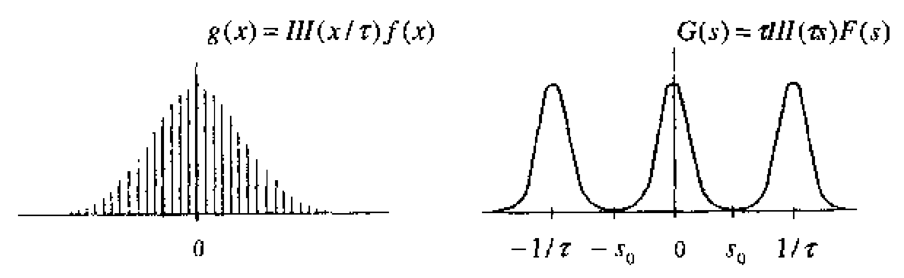


图 5-3 采样后的函数

采样（有时也用Δt 表示采样间隔）：

时域 频域

$$g(x) = \frac{1}{\Delta t} f(x) III(\frac{x}{\Delta t}) \qquad G(s) = \frac{1}{\Delta t} F(s) * \Delta t III(\Delta t s)$$

$$= \frac{1}{\Delta t} F(s) * \sum_{n=-\infty}^{\infty} \delta(s - \frac{n}{\Delta t})$$

$$= \frac{1}{\Delta t} \sum_{n=-\infty}^{\infty} F(s) * \delta(s - \frac{n}{\Delta t})$$

$$= \frac{1}{\Delta t} \sum_{n=-\infty}^{\infty} F(s - \frac{n}{\Delta t})$$

任何用等间隔Δt 采样的函数都有周期为 1/Δt 的频谱。

5.2.2 采样定理

下面将介绍采样定理。在采样过程中，除采样点外，函数其他点的信息都失去了，那么能否从采样点恢复最初的函数呢？如果能够恢复，是否可以完全恢复呢？在数字图像处理中，这是一个非常重要的问题，而采样定理就是从理论上给出一个肯定的回答。

若能从 $G(s)$ 恢复 $F(s)$，则可以从 $g(x)$ 中获得 $f(x)$。其思想就是除了保留中心处于原点的那一个外，消除所有 $F(s)$ 的复制品，如上面图 5-3 所示。

方法：

$$F(s) = G(s)\left[\Delta t \Pi\left(\frac{s}{2f_N}\right)\right] \tag{5.2.2}$$

其中：

$$s_0 \leqslant f_N \leqslant \frac{1}{\Delta t} - s_0 \tag{5.2.3}$$

这里的 f_N 成为折回频率，通常也用 s_1 表示。

则：

$$f(x) = F^{-1}\{F(s)\} = F^{-1}\left\{G(s)\Delta t \prod\left(\frac{s}{2f_N}\right)\right\} \tag{5.2.4}$$

即：

$$f(x)=\Delta t g(x)*2f_N\frac{\sin(2\pi f_N x)}{2\pi f_N x} \tag{5.2.5}$$

由于 $s_0 \leqslant f_N \leqslant \frac{1}{\Delta t}-s_0$，通常取 $f_N=\frac{1}{2\Delta t}$，于是：

$$f(x)=g(x)*\frac{\sin(2\pi f_N x)}{2\pi f_N x}=g(x)*\frac{\sin(\pi x/\Delta t)}{(\pi x/\Delta t)} \tag{5.2.6}$$

从 $g(x)$ 恢复 $f(x)$，只需与内插函数 $c(x)=\frac{\sin x}{x}$ 做卷积。式（5.2.6）说明了确实可以从 $g(x)$ 完全恢复 $f(x)$，但必须受到两个限制：

（1）$f(x)$ 的频谱必须是受限制于 s_0，就是说 $f(x)$ 的有限频谱为 s_0。

（2）采样间隔 τ 和带宽 s_0 之间必须满足式（5.2.3）。

这就引出了著名的采样定理：

以间隔 τ（有时也用 Δt 表示）采样的函数可以完全地从采样数据中恢复，只要：

$$\tau \leqslant \frac{1}{2s_0}\left(\text{或}\frac{1}{\tau}\geqslant 2s_0\right) \tag{5.2.7}$$

其中，s_0 是函数 $f(x)$ 的最高截止频率。

将 $g(x)$ 和式（5.2.6）中的内插函数做卷积，相当于在每个采样点上复制了一个窄的 $\sin(x)/x$ 函数，如前面图 5-3 所示。式（5.2.6）保证了相互重叠的 $\sin(x)/x$ 函数总可以准确地恢复原函数。

5.2.3 欠采样与混叠效应

采样定理决定了采样时要尽可能地密集采样，得到足够的采样信息，这样才能从采样值中完全恢复原图像。这个思想在哲学上也是比较显然的，如果需要重构复杂的信息，首先要有足够的信息来支撑，但有时对某些高频信息并不能做到密集采样，如果采样定理不能满足，将会有什么样的结果呢？下面将讨论这样的情况。

假定 $\tau>\frac{1}{2s_0}$。那么当 $F(s)$ 被复制以形成 $G(s)$ 时，每个复制品会重叠在一起，有的是高频相加（如图 5-4 所示），有的是抵消。如果还是用式（5.2.6）中的函数内插，则不可能准确地恢复 $f(x)$。因为：

$$G(s)\Pi\left(\frac{s}{2s_1}\right)\neq F(s) \tag{5.2.8}$$

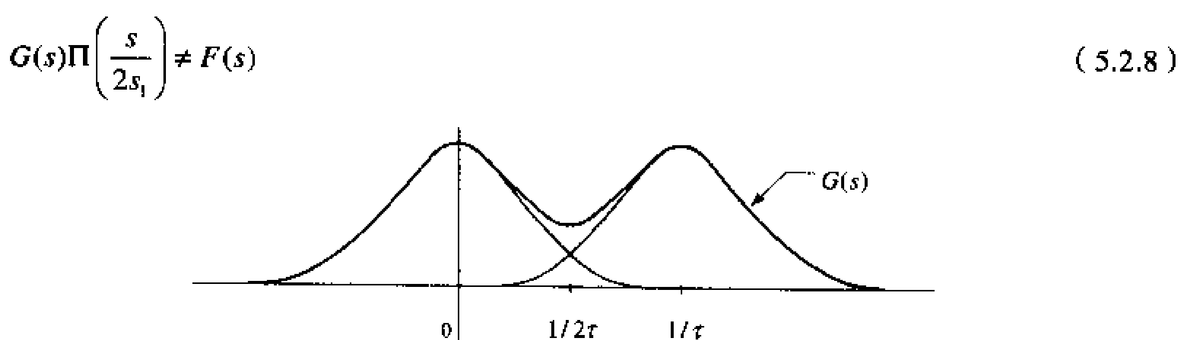


图 5-4 频谱复制区间的重叠

频谱复制品重叠的影响可以这样来解释：频率 s_1 以上的能量被折叠返回到 s_1 以下并且

被加到频谱上。这种能量的折回被称作“混叠”，$f(x)$ 和内插所得到的函数之间的差别叫做“混叠误差”。

总的来说，s_1 以上频率有越多的能量，越多的能量就会折回到频谱中，混叠误差就越严重。当 $f(x)$ 为偶函数时，$F(s)$ 也是偶函数，混叠的效果是提高了频谱中的能量；如果 $f(x)$ 是奇函数，发生相反的事情，频谱中的能量减少。如果 $f(x)$ 非奇非偶，则混叠增大了偶函数部分，降低了奇函数部分，使得函数和其频谱比实际上更加接近偶函数。

5.2.4 内插

将 $g(x)$ 和等式（5.2.5）所建议的内插函数卷积在效果上等于在每个采样点上复制一个窄的 $\sin(x)/x$ 函数，如图 5-5 所示。等式（5.2.5）保证了相互重叠的 $\sin(x)/x$ 函数的总和可准确地回复原函数。

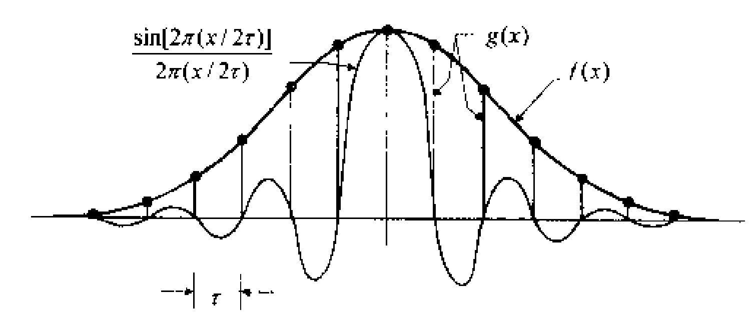


图 5-5　用内插函数恢复原函数

如果采样间隔的倒数明显地大于带宽 s_0，等式（5.2.3）允许在 $\sin(x)/x$ 的频率选择上有相当大的自由度：可以将 f_0 设为 s_0 和 $1/\tau - s_0$ 的任何值。为了方便起见，可将 f_0 放在中点：

$$f_0 = \frac{1}{2\tau} \tag{5.2.9}$$

这样内插函数为：

$$\frac{1}{\tau}\frac{\sin(\pi x/\tau)}{\pi x/\tau} \tag{5.2.10}$$

5.2.5 采样与欠采样分析

下面分析几种采样情况来加深对采样定理的了解。假定函数：

$$f(t) = 2\cos(2\pi f_0 t) \qquad T = \frac{1}{f_0} \tag{5.2.11}$$

其频谱函数为：

$$F(s) = \delta(s + f_0) + \delta(s - f_0) \tag{5.2.12}$$

如图 5-6 所示。在假定以相等间隔 Δt 对 $f(t)$ 进行采样。$f(t)$ 周期为 $1/f_0$。

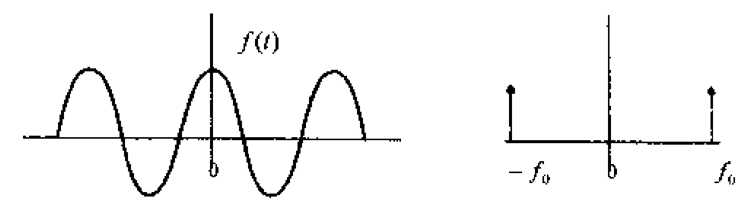


图 5-6　余弦函数及其频谱

情形 1：过采样。

假定采样间隔：

$$\Delta t = \frac{1}{4}(\frac{1}{f_0}) < \frac{1}{2f_0} \tag{5.2.13}$$

折回频率：

$$f_N = \frac{1}{2\Delta t} = \frac{1}{2\tau} = 2f_0 \tag{5.2.14}$$

$F(s)$ 在高于 f_N 之后没有能量，$f(t)$ 可完全恢复。如图 5-7 所示。

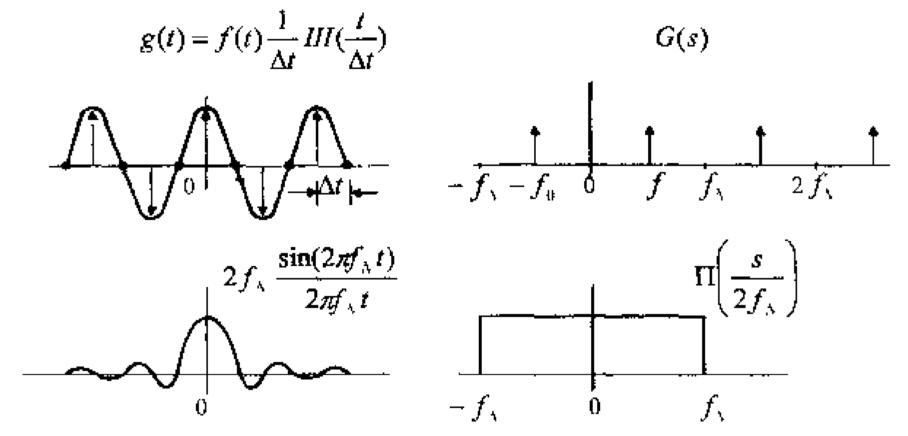


图 5-7 采样余弦函数，情形一

情形 2：临界采样。

假设：

$$\Delta t = \frac{1}{2}(\frac{1}{f_0}) \tag{5.2.15}$$

则：

$$f_N = f_0 \tag{5.2.16}$$

并且每个周期采样两个点。图 5-8 表示临界情况。这里采样的是余弦函数的正负峰值点，而函数仍然可以通过内插完全恢复。

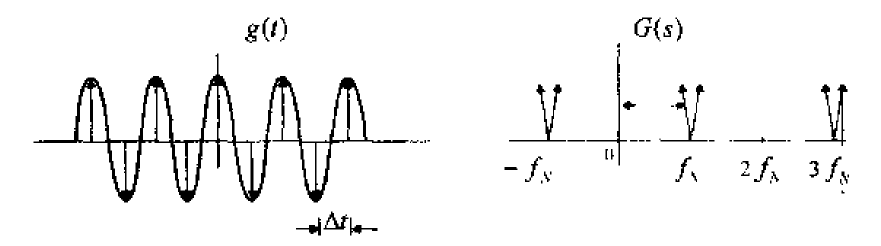


图 5-8 采样余弦函数，情形二

在邻域中，相邻复制品的冲激在 $s = f_0$ 处合到一起，但内插函数的频域在该点取值 1/2，所以函数仍可以无失真地恢复。

情形 3：欠采样。

$$\Delta t = \frac{2}{3}(\frac{1}{f_0}) > \frac{1}{2f_0} \tag{5.2.17}$$

$$f_N = \frac{3}{4}f_0 \tag{5.2.18}$$

中心为 $S=2f_N$ 的复制品的左侧落在 $S={}^{f_0}\!/\!_2$ 处，产生混叠。

重建后的函数为：$f(t)=2\cos(\pi f_0 t)$

情形 4：严重欠采样。

$$\Delta t=(\frac{1}{f_0}) \tag{5.2.19}$$

$$f_N=\frac{1}{2}f_0 \tag{5.2.20}$$

频率 f_0 处的能量混叠到频率 0 处。事实上，余弦仅在正峰值被采样，内插后为常数 1。

情形 5：和情况 2 一样，只是将输入函数改为正弦函数。

$$f(t)=2\sin(2\pi f_0 t) \tag{5.2.21}$$

$$F(s)=j\left[\delta(s+f_0)-\delta(s-f_0)\right] \tag{5.2.22}$$

若：

$$\Delta t=\frac{1}{2}(\frac{1}{f_0})\text{，则 } f_N=f_0 \tag{5.2.23}$$

相邻频域复制区间重叠的奇冲激对在 $S=f_N$ 处重叠并抵消。重建后函数为：$f(t)=0$。

5.3　频谱计算

5.3.1　时域中截取

假设对于一段 T 长度的信号 $f(t)$，用间隔为 Δt 的采样点采样，共有 N 个采样点：

$$N=\frac{T}{\Delta t} \tag{5.2.24}$$

其中 T 为截取窗口的宽度。因为一个信号只能用有限数目的点来采样，因此采样过程忽略了信号在截取窗口外面的部分，这等同于将截取窗口外面的信号置零。

如果要用 $f(t)$ 的采样值来计算频谱 $F(s)$ 上的点，可以通过将傅立叶变换编程为一个数字积分来实现，也就是离散傅立叶变换。首先必须决定在频谱上计算所有的点数、采样点的间距，以及计算的频谱范围。

既然采样后的信号包含 N 个相互独立的测量，在频谱上进行总共 N 个点的计算也是合理的。计算更多的点会导致冗余，计算更少的点则会使有关的所有信息不能使用。因此，一个通用的计算傅立叶变换的程序应该从 N 个（复数）采样点计算出频谱上的 N 个（复数）点。为了方便起见，通常使得计算的点在 s 轴上等间距地分布。

5.3.2　频域上截取

既然 $f(t)$ 是以采样间隔 Δt 采样后的函数，其频谱是周期函数，第二个周期的中心为 $\frac{1}{\Delta t}$，故主频率 $(F(s))$ 变化范围为：

$$-\frac{1}{2\Delta t}\leqslant s\leqslant\frac{1}{2\Delta t} \tag{5.2.25}$$

其中：

$$s_m = \frac{1}{2\Delta t} \tag{5.2.26}$$

则 $F(s)$ 上的采样间隔为：

$$\Delta s = \frac{2S_m}{N} = \frac{1}{N\Delta t} = \frac{1}{T} \tag{5.2.27}$$

即：一个域的采样间隔与另一个域的截取窗口成反比。

5.3.3 频谱计算

简言之，一个域中的采样间隔决定了（或决定于）另一个域中的截取窗口的宽度。如果要计算频谱中的高频成分，则必须在频域中密集地采样。如果要频谱中的高分辨率（小的 Δs），则必须在时域中采用大的截取窗口，即使时域函数很窄。表 5-1 中列出了窄时域和频域中采样和截取的参数之间的关系。

表 5-1 窄时域和频域中采样和截取的参数之间的关系

参数	域	关系
采样点数	时域，频域	$N = \frac{T}{\Delta T} = \frac{2s_m}{\Delta s}$
采样间隔	时域	$\Delta t = \frac{T}{N} = \frac{1}{2s_m}$
采样间隔	频域	$\Delta s = \frac{2s_m}{N} = \frac{1}{N}$
截断窗口宽度	时域	$T = N\Delta t$
最大可计算频率（即 Nyquist 频率或折回频率）	频域	$s_m = \frac{1}{2\Delta t} = \frac{1}{2}N\Delta s$

如果要计算其频谱的 $f(t)$ 是复函数，则 N 个实部值和 N 个虚部值变换后产生频谱中的 N 个实值和 N 个虚值。如果 $f(t)$ 是实值函数，则 N 个实值和 N 个 0（虚部）将在频谱的右半边产生 $N/2$ 个实数值和 $N/2$ 个虚数值。由于 $F(s)$ 是 Hermite 型的，频谱的左半部分是右半部分的镜像。这样从信息内容的角度来看，频谱的左半部分的 $N/2$ 个实值和 $N/2$ 虚值是冗余的。

注意：这两个情况下，未受限的采样点的数目在两个域中是一样的。

5.4 混叠

5.4.1 混叠的不可避免性

被处理的大多数信号是带宽有限的，但在时间域上信号 $f(t)$ 是无限延伸的。人们的处理实际上仅取一段长度为 2a 的信号，即取：

$$g(t) = f(t)\cdot\prod\left(\frac{t}{2a}\right) \tag{5.4.1}$$

等价于：

$$G(s) = F(s) * 2a\frac{\sin 2\pi as}{2\pi as} \tag{5.4.2}$$

其频谱是非有限性的。即截取破坏了带宽有限性，也就是说数字处理在任何情况下都

造成混叠，但是误差可被限定。

采样定理指出，在对一带宽有限的函数采样时，选用合适的采样间距可以完全避免混叠。这样，如果带宽有限的函数时，则可以避免混叠。但另一方面，如果不得不处理带宽非有限的函数，则必须在混叠的干扰下工作。不过，现实中人们遇到的几乎都是带宽无限的函数，所以必须受到截取过程的阻扰。假定一个带宽有限的函数被截取了一段有限长度 T，这个过程可模型化为将函数与宽度为 T 的矩形脉冲相乘。这个过程等价于将其频谱与无限连续延伸的 $\frac{\sin(x)}{x}$ 函数在频域上卷积。

因为两个函数结果不可能比其中任意一个窄，所以，经过截取的函数的频谱在频域内是无限延伸的。可见，截取破坏了带宽的有限性，注定了数字图像处理在任何时候都造成混叠。尽管混叠不能完全避免，产生的误差却可以被限定，并且可以被降低到实际应用可以接受的程度。

5.4.2 混叠误差的上界

假定输入信号为：

$$f(t)=\frac{1}{2a}\prod(\frac{1}{2a}) \tag{5.4.3}$$

其频谱函数为：

$$F(s)=\frac{\sin(2a\pi s)}{2a\pi s} \tag{5.4.4}$$

截取窗口为 T。

考虑 s_m 处的混叠，则：

$$A=|F(s_m)|\leqslant\frac{1}{2\pi as_m}=\frac{2\Delta t}{2\pi a}=\frac{\Delta t}{\pi a} \tag{5.4.5}$$

其中：

$$s_m=\frac{1}{2\Delta t} \tag{5.4.6}$$

则混叠误差上界与 Δt 成正比，但与 T 无关。只要 Δt 与 2a 比足够小，则可使混叠误差小到所需的程度。

5.4.3 频谱分辨率

采样点数越密，分辨率越高。

$F(s)$ 具有频率为 a 的正弦波，用 M 表示为在所计算的频谱 $F(s)$ 上每个周期的采样点数，并用它作为频谱分辨率的度量。参数 M 表示如何精密地对采样频谱 $F(s)$ 进行计算。

Δs 为采样间隔，$F(s)$ 地正弦波动周期是 $1/a$，则：

$$M\Delta s=\frac{1}{a} \tag{5.4.7}$$

$$M=\frac{1}{a\Delta s}=\frac{T}{a} \tag{5.4.8}$$

其中：

$$\Delta s=\frac{1}{T} \tag{5.4.9}$$

想要 M 大，必须 T 大。即，要同时取得小的混叠误差和高的频率分辨率，则 Δt 小，T

大，从而采样点数大，计算量大。

5.5 截取

计算一个边缘的频谱（近似地用阶跃函数），这个技术通常用来判定一个作用包含边缘的图像滤波器的传递函数。由于对边缘的截取是一个非常重要的变化，所以下面来介绍截取的影响。定义阶跃函数如下：

$$f(x)=sign(x) \tag{5.5.1}$$

$$F(s)=\frac{-j}{\pi s} \tag{5.5.2}$$

阶跃函数是以常量向两边无限延伸的，所以对其截取将会是十分敏感的。阶跃函数及其频谱如图 5-9 所示。

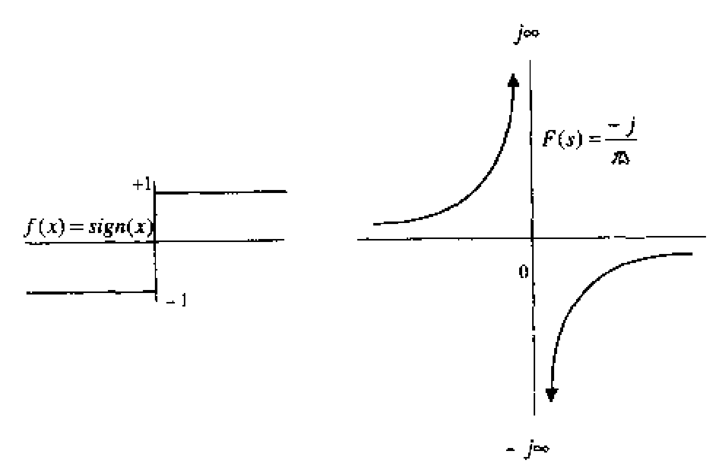


图 5-9 阶跃函数及其频谱

用宽度为 T 窗口对 $f(x)$ 进行截取，得到函数 $g(x)$：

$$\begin{aligned} g(x)&=f(x)\Pi(\frac{x}{T}) \\ &=\left[\Pi(\frac{x-T/4}{T/2})-\Pi(\frac{x+T/4}{T/2})\right] \\ &=\Pi(\frac{x}{T/2})*\left[\delta(x-\frac{T}{4})-\delta(x+\frac{T}{4})\right] \end{aligned} \tag{5.5.3}$$

于是其频谱为：

$$G(s)=-2j\sin(\pi s\frac{T}{2})\frac{\sin(\pi sT/2)}{\pi s} \tag{5.5.4}$$

即：

$$G(s)=\frac{-2j}{\pi s}\sin^2(\pi s\frac{T}{2}) \tag{5.5.5}$$

最后得：

$$G(i)=2F(i)\left[\frac{1}{2}-\frac{1}{2}\cos(\pi iT)\right] \tag{5.5.6}$$

如图 5-10 所示，截取后信号的频谱是被两倍于所要求的频谱 $F(s)$ 所包络的正弦曲线。频谱的巨大变化的原因就是因为截取，敏感性在这里体现出来。

由于实际上做的是计算 $G(s)$ 上的点，所以可以得到截取效应的表示：

$$G(i)=\begin{cases}2F(i) & i\text{奇数}\\ 0 & i\text{偶数}\end{cases} \tag{5.5.7}$$

如图 5-11 所示。

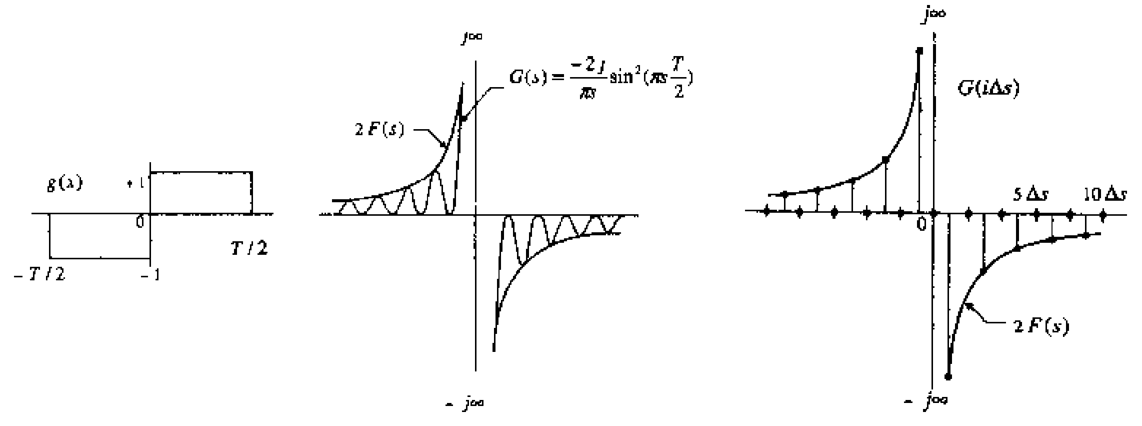


图 5-10 截取后的阶跃函数及其频谱　　　　图 5-11 阶跃函数的计算频谱

截取效应：

注意截取在上例中的奇怪而有趣的效应：奇数号点是正确的（尽管是正常大小的两倍），而偶数号点为 0。这表明截取在奇数号点和偶数号点之间重新分配了能量。

在本例中，边缘在截取窗口的中央，请读者考虑：当边缘稍微偏离截取窗口的中央时，$G(s)$ 上的采样点受到的影响。

通过 $G(i\Delta s)$ 与一个窄的单调性局部平均滤波器如[1/4，1/2，1/4]相卷积，可获得所期望的结果。这等价于将截取后的边缘与一个形如 $(\sin(x)/x)^2$ 的窗口函数相乘的结果。这样可以避免在 $\pm T/2$ 出现的不连续，从而避免截取误差。

5.6 数字处理简介

考察对一个连续信号进行数字处理带来的整个累积影响，即截取、采样、内插、数字化卷积和傅立叶变换的影响。在本节中，只数字化一个函数，然后在不经过处理的情况下重构它。这里用图 5-12 所示的一个连续函数 $f(t)$ 来开始讨论，差函数具有三角形的幅度谱，但是具有随机相位。

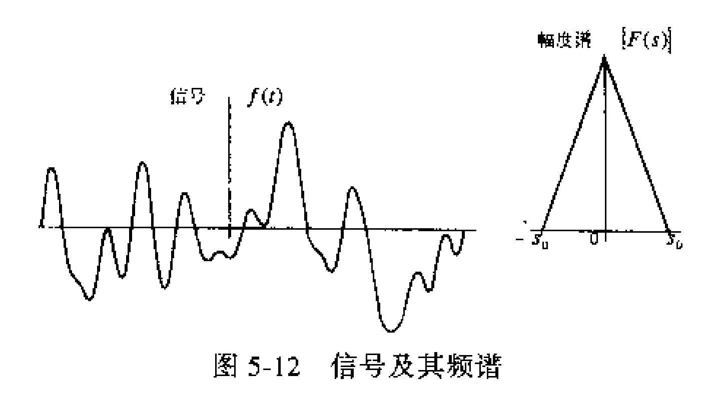
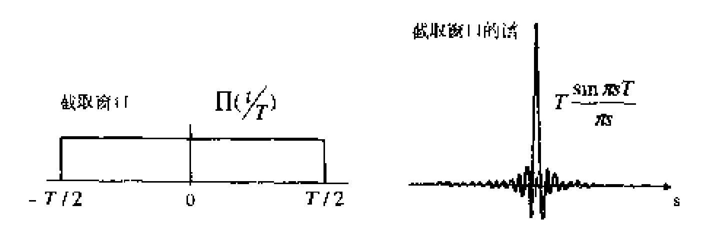


图 5-12 信号及其频谱

1. 截取

当数字化该信号时，必须在有限区间 T 内对其进行截取。截取窗口 $\Pi(t/T)$ 及其频谱如

图 5-13 所示。图中还给出了截取后的函数及其频谱。对 $f(t)$ 的截取将使其频谱与一个窄的 $\sin(x)/x$ 卷积。



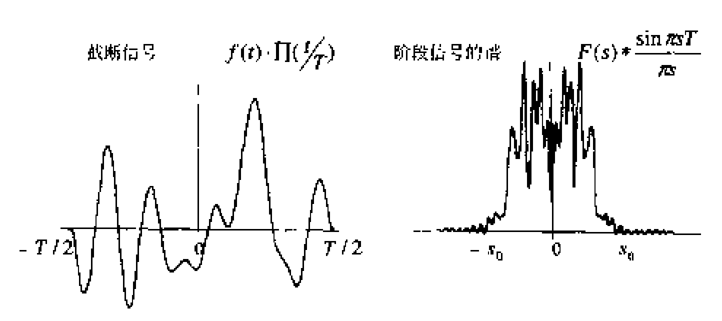


图 5-13　信号的截取

$$g_1(t)=f(t)\cdot\Pi(t/T) \tag{5.6.1}$$

$$G_1(s)=F(s)*\frac{\sin \pi sT}{\pi s} \tag{5.6.2}$$

2. 采样孔径

数字化器相当于每个采样点上有一有限宽度的采样孔径。

$$g_2(t)=g_1(t)*\frac{1}{\tau}\Pi(\frac{t}{\tau}) \tag{5.6.3}$$

$$G_2(s)=G_1(s)\frac{\sin \pi s\tau}{\pi \tau s} \tag{5.6.4}$$

3. 采样

$$g_3(t)=g_2(t)\frac{1}{\Delta t}III(t/\Delta t) \tag{5.6.5}$$

$$G_3(s)=G_2(s)*III(s\Delta t) \tag{5.6.6}$$

4. 内插

理想的内插是将位于 $s=0$ 的主复制品 $F(s)$ 与 $\Pi(\frac{t}{2s_1})$ 乘积，相当于：

$$g_3(t)*\left[2s_1\frac{\sin(2\pi s_1 t)}{2\pi s_1 x}\right] \tag{5.6.7}$$

若用 $G_3(s)$ 与 $\left[\frac{\sin(\pi s t_0)}{\pi s t_0}\right]^2$ 乘积相当于：

$$h(t)=g_3(t)*\frac{1}{t_0}\Lambda(\frac{t}{t_0}) \tag{5.6.8}$$

故：

$$h(t)=\left(\left\{\left[f(t)\Pi(\frac{t}{T})\right]*\frac{1}{\tau}\Pi(\frac{t}{\tau})\right\}\frac{1}{\Delta t}III(\frac{t}{\Delta t})\right)*\frac{1}{t_0}\Lambda(\frac{t}{t_0}) \tag{5.6.9}$$

$$H(s)=\left(\left\{\left[F(s)*T\frac{\sin(\pi sT)}{\pi sT}\right]\frac{\sin(\pi s\tau)}{\pi s\tau}\right\}*III(s\Delta t)\right)\left[\frac{\sin(\pi st_0)}{\pi st_0}\right]^2 \tag{5.6.10}$$

数字化后误差是明显的。

5. 数字处理的影响

数字化对信号肯定有影响，影响有多大?

（1）通常选择：

$\tau=t_0=2\Delta t$（对于矩形函数的插值，$\tau=2t_0=2\Delta t$）。

（2）截取使频谱与 $T\frac{\sin(\pi Ts)}{\pi Ts}$ 相卷积，T 越大，频谱越窄，混叠影响越小。

（3）采样孔径能降低频谱中的高频能量，可减少混叠。

（4）采样使频谱变成周期性，导致折叠频率 $1/2\Delta t$ 以上能量产生混叠。

（5）内插：

理想的内插函数为 $\sin x/x$，其他内插函数不能消除频谱的其他复制品，还可能减少主复制品的能量。

5.7 混叠误差的控制

选择采样孔径和采样间隔可以控制混叠。

1. 采样孔径

选择 $\tau=2\Delta t$ 的矩形脉冲作采样孔径，使其传递函数的第一个过零点于 $f_N=1/2\Delta t$，f_N 以上频率的能量会被大大减少。

选择 $\tau=4\Delta t$ 的三角采样孔径，第一个过零点也为 f_N，其频率下降更快，能有效减少混叠。

2. 过采样

一个连续函数数字化处理必然出现严重畸变，但可以通过过采样减少误差。

采用 Δt 应尽可能小，因为 Δt 越小，越使 f_N 落在有用的频率之外，即使混叠污染了频谱的高频部分，对有用的数据也影响不大。

小结

本章主要介绍了预备知识、采样和插值、频谱计算、混叠、截取、数字处理简介、混叠误差的控制等内容。

重点：本章重点主要有以下几个方面：

（1）Shah 函数（冲激序列）是其自身的傅立叶变换。

（2）将 Shah 函数拉伸和压缩会改变其冲激强度（高度）。

（3）对连续函数的采样可以模型化为 Shah 函数相乘。

（4）如果采样点间距不大于 $1/2s_0$，一个频谱带宽为 s_0 的函数可以从它的采样值中完全恢复（采样定理）。

（5）欠采样会引起混叠，折叠频率（$s=1/2\Delta t$）以上的能量会在折叠频率以下的相同距离内出现。

（6）截取破坏了频谱的带宽有限性，所以在数字处理中误差不可避免，但是可以控制。

（7）尽量过采样或者在采样前进行低通滤波器滤波可以降低混叠的影响。

难点： 如何根据采样定理进行采样和插值，控制混叠的误差。

疑点： 混叠是否可以避免，如何避免。

练习五

一、选择题

1. 假定函数为 $f(x)=2\cos(2\pi f_0 t)$，其频谱为 $F(s)=\delta(s+f_0)+\delta(s-f_0)$，再假定以相等间隔 Δt 对 $f(t)$ 采样，其中 $f(t)$ 的周期是 $1/f_0$。问 Δt 等于多少时进行采样不会产生混叠，能够完全恢复（　）。

   A. $1/f_0$　　B. $2/(3f_0)$　　C. $1/(4f_0)$　　D. $3/(5f_0)$

2. 一个以间隔 t 采样的函数可以完全地从采样数据中恢复，只要（　）。

   A. $\tau\leqslant\frac{1}{2s_0}$　　B. $\tau<\frac{1}{2s_0}$　　C. $\tau\leqslant\frac{1}{s_0}$　　D. $\tau<\frac{1}{s_0}$

3. 对带宽有限的函数采样时，过采样是否会产生混叠（　）。

   A. 会　　B. 不会　　C. 一般情况下会　　D. 无法判断

二、思考题

1. 一周期信号频率为 $f_0=3Hz$。要通过计算其频谱来确定它的谐波频率有多高（它已通过一个截止频率为 $48Hz$ 的低通滤波器）。如果用临界采样数字化该信号，最小的采样点个数是多少，在多长的时间内？

2. 一个信号是频率 $f_0=0.22Hz$ 的余弦函数，被一幅值为 4，标准差 $\sigma=10$ 秒，中心在 $t=0$ 的高斯曲线所包络，为实现截取，认为当其幅度降到峰值的 0.1％时信号就是零了。为数字化该信号需要在多大范围内进行多少次采样：

（1）临界采样？

（2）两倍采样？

（3）如使用一总是进行 256 个采样化数字化器，还会出现混叠吗？

三、上机题

设计一个程序，对一幅噪音图像进行数字处理，去除噪音。

第 6 章　离散图像变换

图像是连续的，但是由于在计算机上对图像进行数字处理时必然受到计算机的限制，而只能对连续图像的离散表达形式进行处理，所以许多数字图像处理步骤要求在处理这些离散数据时必须考虑到采样和插值的问题。

在图像压缩中，人们希望将一幅图像以更紧凑的数据格式进行编码，同时保持信息不丢失或者仅仅丢失能够容忍的一小部分。而离散图像变换使得图像压缩有了很大的发展空间。

数字图像可以当作一个数据文件来看待。

本章主要内容有：

（1）线性变换。

（2）基图像和基函数。

（3）正弦型变换和方波型变换。

（4）基于特征向量的变换。

（5）变换域滤波。

6.1　线性变换

变换是图像特征提取和图像压缩的主要手段。本章主要了解变换的矩阵运算，因为图像以矩阵的形式存在，图像的运算即是矩阵的运算。

离散图像的表示：一个图像可以看做 $M\times N$ 样本的矩阵，可用矩阵 f 表示。

$$f=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix}$$

也可以看做向量 f：

$$f=[f(0,0),f(0,1),\cdots,f(0,N-1),f(1,0),f(1,1),\cdots f(1,N-1),\cdots,\\ f(M-1,0),f(M-1,1),\cdots,f(M-1,N-1)]$$

6.1.1　一维离散线性变换

设 x 为 $N\times 1$ 向量，T 是 $N\times N$ 矩阵，则线性变换为：

$$y=Tx \tag{6.1.1}$$

x 与 T 每行内积，即：

$$y_i=\sum_{j=0}^{N-1}t_{i,j}x_j \tag{6.1.2}$$

其中，T 为核矩阵。上两式等价定义了一个线性变换，变换得到的是另一个 $N\times 1$ 的向量 y。

由于 y 是由输入元素的一阶和构成，所以将其称作线性变换。y 向量中第 i 个分量元

素 y_i 是输入向量 x 和核矩阵 T 的第 i 行的内积。通常比较常见的线性变换有平移、伸缩、旋转、FFT 等等。其中一个比较简单的例子就是二维坐标系统中的一个向量的旋转，令该矩阵：

$$T=\begin{bmatrix}\cos(\theta) & -\sin(\theta)\\ \sin(\theta) & \cos(\theta)\end{bmatrix} \tag{6.1.3}$$

则：

$$\begin{bmatrix}y_1\\ y_2\end{bmatrix}=y=Tx=\begin{bmatrix}\cos(\theta) & -\sin(\theta)\\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} \tag{6.1.4}$$

此变换将向量 x 旋转了 θ 角，得到旋转后的向量 y。

若 T 非奇异，则：

$$x=T^{-1}y \tag{6.1.5}$$

这里相对于前一个例子来说，相当于将 y 反相旋转了 θ 角恢复到 x。

6.1.2 酉变换

首先定义一个比较特殊的变换。

定义： 若核矩阵 T 为酉矩阵，则称由 T 定义的线性变换叫做酉变换。

（1）变换矩阵称为酉矩阵。

如果：

$$T^{-1}=T^{*t} \tag{6.1.6}$$

即：

$$TT^{*t}=T^{*t}T=I \tag{6.1.7}$$

其中 T^{*T} 表示 T 的共轭转置。

（2）若酉矩阵 T 为实矩阵，则称 T 为正交矩阵，这时：

$$TT^{t}=T^{t}T=I \tag{6.1.8}$$

TT^{T} 的第（i,j）元素为 T 的第 i 和第 j 行内积。$i=j$ 时为 1；$i\neq j$ 时为 0。

即：

$$T(i,:)*T(j,:)=\delta_{ij}=\begin{cases}1 & i=j\\ 0 & i\neq j\end{cases} \tag{6.1.9}$$

（3）酉矩阵的例子：离散傅立叶变换（DFT）。

$$F=Wf \tag{6.1.10}$$

即：

$$F_k=\frac{1}{\sqrt{N}}\sum_{i=1}^{N-1}f_i\exp(-j2\pi k\frac{i}{N}) \tag{6.1.11}$$

其中 w 是一个酉矩阵（但不是正交矩阵），其元素为：

$$w_{ik}=\frac{1}{\sqrt{N}}\exp(-j2\pi k\frac{i}{N}) \tag{6.1.12}$$

（4）物理意义（$y=Tx$）。

酉变换所有行定义了一组正交基，对 x 做变换，获得 x 在正交基上的投影，变换系数规定了在原信号中，各分量占有的量；另一方面，反变换代表合成过程，变换系数规定了

为精确、完全地重构输入向量而加人的各分量的大小。如图 6-1 所示。

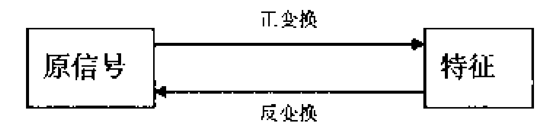


图 6-1 酉变换的物理意义

通常，核矩阵 T 是非奇异的（即 T 的秩为 N），这就使得变换可逆。通常正变换被看成一个分解过程：将信号分量分解成它的各个基元分量，这些基元分量自然以基向量的形式表示。变换系数如前说明一样，规定了在原信号中，各分量所占有的量。

另一方面，直观上来说，反变换通常是一个合成过程，通过将各个分量相加来合成原始分量。

这个过程的原理支撑为：任何一个向量都能惟一地分解成分别具有合适地幅度地一组基向量，然后通过将这些分量相加可以重构原向量。当然，变换系数的个数和向量的元素个数是相等的，这样在变换前后的自由度数目是相等的，从而保证这个变换没有引人新的信息，也未破坏任何原有的信息。

6.1.3 二维离散线性变换

定义 $N\times N$ 矩阵 $F(i,k)$ 到 $N\times N$ 矩阵 $G(m,n)$ 的变换为：

$$G_{mn}=\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}F_{ik}\psi(i,k,m,n) \tag{6.1.13}$$

其中，$\psi(i,k,m,n)$ 是变换的核矩阵，为 $N^2\times N^2$ 矩阵。

我们称核可分离，或变换可分离，当且仅当：

$$\psi(i,k,m,n)=T_r(i,m)T_c(k,n) \tag{6.1.14}$$

例：

$$\begin{aligned}w_{ikmn}&=\frac{1}{N}\exp\left[-j2\pi(\frac{im}{N}+\frac{kn}{N})\right]\\&=\frac{1}{\sqrt{N}}\exp(-j2\pi\frac{im}{N})\cdot\frac{1}{\sqrt{N}}\exp(-j2\pi\frac{kn}{N})\end{aligned} \tag{6.1.15}$$

若：

$$T_c(i,k)=T_r(i,k) \tag{6.1.16}$$

则称变换为对称的，这时式（6.1.14）变为：

$$\psi(i,k,m,n)=T(i,m)T(k,n) \tag{6.1.17}$$

式（6.1.13）也可写为：

$$G_{mn}=\sum_{i=0}^{N-1}T(i,m)\left[\sum_{k=0}^{N-1}F_{i,k}T(k,n)\right] \tag{6.1.18}$$

即：

$$G=TFT \tag{6.1.19}$$

对于二维可分离的、对称的酉变换。反变换为：

$$F=T^{-1}GT^{-1}=T^{*T}GT^{*T} \tag{6.1.20}$$

如果矩阵 T 为实对称，则：

$$F = TGT \tag{6.1.21}$$

例：二维 DFT，核矩阵 W 为对称矩阵：

$$G = WFW \tag{6.1.22}$$

DFT 的反变换使用 $W^{-1} = W^{*t}$ ，又 W 为对称矩阵，所以 $W^{*t} = W^{*}$ ，则：

$$F = W^{*}GW^{*} \tag{6.1.23}$$

6.2 基函数和基图像

6.2.1 基函数

核矩阵各行构成了 N 维向量空间的一组基向量。这些行都是正交的，即满足：

$$TT^{*t} = T \tag{6.2.1}$$

或者：

$$\sum_{i=0}^{N-1} T_{j,i} T_{k,i}^{*} = \delta_{j,k} \tag{6.2.2}$$

其中 $\delta_{j,k}$ （Kronecker δ）如式（6.1.9）定义。

虽然任何一组正交向量集都可以用于一个线性变换，然而，通常整个集皆取自同一种形式的基函数，例如，傅立叶变换就是使用复指数作为其基函数的原型。各个基函数之间只是频率不同而已。

空间中任何一个向量都可以用单位长度的基向量的加权和来表示（合成）。任何一个一维的酉变换都对应着 N 维向量空间的一个旋转。另外，对于一个 $N \times N$ 的图像矩阵可以通过行堆积构成一个 $N^2 \times 1$ 的向量，则任意一个二维的、对称的、可分离的酉变换对应着一个 N^2 维的向量空降中向量的旋转。

6.2.2 基图像

前面已经解释过，反变换通常是一个合成过程，通过将各个分量相加来合成原始分量。所以二维反变换可以看作是通过一组被适当加权的基图像求和而重构原图像。变换矩阵 G 中的每个元素就是其对应的基本图像在求和时所乘的倍数。

对于变换 $F = TGT$ ，令：

$$G = G^{pq} = \{\delta_{i-p,k-q}\} \tag{6.2.3}$$

产生一个基图像。其中 G 是一个只含有一个非零元素 1 的系数矩阵。i 和 j 分别是行和列的下标，p 和 q 分别是 G 中非零元素的行和列的位置。则反变换为：

$$J_{m,n} = TG^{m,n}T = \sum_{i=0}^{N-1} T(i,m)\left[\sum_{k=0}^{N-1} \delta_{i-p,k-q} T(n,k)\right] = T(p,n)T(m,q) \tag{6.2.4}$$

共 N^2 个基图像。

$$F = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} G(m,n) J_{m,n} \tag{6.2.5}$$

其中 $J_{m,n} = TG^{m,n}T$ 。

若核矩阵 T 的各行时正交的，即：

$$TT^{*t} = I \tag{6.2.6}$$

或：

$$\sum_{i=0}^{N-1} T_{ji} T_{ki}^{*} = \delta_{j,k} \tag{6.2.7}$$

则称 T 的每一行为一个基函数。

6.3 正弦型变换和方波型变换

本节将介绍一些常用的变换。关键是变换核的不同。

6.3.1 正弦型变换

1. 离散傅立叶变换

主要回忆一下 DCT 的核矩阵 W：

$$W = \begin{bmatrix} w_{0,0} & \cdots & w_{0,N-1} \\ \cdots & \cdots & \cdots \\ w_{N-1,0} & \cdots & w_{N-1,N-1} \end{bmatrix} \tag{6.3.1}$$

其中：

$$w_{i,k} = \frac{1}{\sqrt{N}} e^{-j2\pi\frac{ik}{N}} \tag{6.3.2}$$

由于虚指数的周期性，W 是酉矩阵。

一维时，DFT 的正变换和反变换分别是：

$$F = Wf \text{ 和 } f = W^{*T} F \tag{6.3.3}$$

注意：在实验中，采用 SHIFT 方法，通常将变换的原点移到 $\left(\frac{N}{2},\frac{N}{2}\right)$ 处，以看到完全的频谱，由周期和位移特性可以推出：

$$f(x,y)(-1)^{x+y} \Leftrightarrow F\left(u-\frac{N}{2}, v-\frac{N}{2}\right) \tag{6.3.4}$$

2. 离散余弦变换（DCT）

二维离散余弦变换（DCT）的定义：

$$G_c(m,n) = \alpha(m)\alpha(n)\sum_{i=0}^{N-1}\sum_{k=0}^{N-1} g(i,k)\cos\left[\frac{\pi(2i+1)m}{2N}\right]\cos\left[\frac{\pi(2k+1)n}{2N}\right] \tag{6.3.5}$$

它的反变换为：

$$g(i,k) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} \alpha(m)\alpha(n) G_c(m,n)\cos\left[\frac{\pi(2i+1)m}{2N}\right]\cos\left[\frac{\pi(2k+1)n}{2N}\right] \tag{6.3.6}$$

其中，系数为：

$$\alpha(0) = \sqrt{\frac{1}{N}}, \text{且} \alpha(m) = \sqrt{\frac{2}{N}}, \quad 1 \leqslant m \leqslant N \tag{6.3.7}$$

表示成酉矩阵运算：

$$G_c = CgC \tag{6.3.8}$$

其中，核矩阵的元素为：

$$C_{i,m} = \alpha(m)\cos\left[\frac{\pi(2i+1)m}{2N}\right] \tag{6.3.9}$$

N 个点的离散余弦变换可以看作 2N 个点的离散傅立叶变换来计算：

$$F(u)=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1}f(x)\cos\frac{(2x+1)u\pi}{2N}$$
$$=\sqrt{\frac{2}{N}}\sum_{x=0}^{N-1}f(x)\mathrm{Re}\left(\exp\left(-\frac{(2x+1)u\pi}{2N}\right)\right) \tag{6.3.10}$$
$$=\sqrt{\frac{2}{N}}\mathrm{Re}\{\sum_{x=0}^{N-1}f(x)\exp\left(-\frac{(2x+1)u\pi}{2N}\right)\}$$

将 $f(x)$ 延拓到 $f_e(x)$，其中：

$$f_e(x)=\begin{cases}f(x) & x=0,1,\cdots,\cdots,N-1\\ 0 & x=N,N+1,\cdots,2N-1\end{cases} \tag{6.3.11}$$

则 $f_e(x)$ 的离散余弦变换为：

$$F(0)=\sqrt{\frac{1}{N}}\sum_{x=0}^{2N-1}f_e(x) \tag{6.3.12}$$

$$F(m)=\sqrt{\frac{2}{N}}\mathrm{Re}\left\{\exp(-\frac{ju\pi}{2N})\sum_{x=0}^{2N-1}f_e(x)\exp(-\frac{2xu\pi}{2N})\right\} \tag{6.3.13}$$

3. 离散正弦变换（DST）

离散正弦变换的定义如下：

$$G_s(m,n)=\frac{2}{N+1}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}g(i,k)\sin\left[\frac{\pi(i+1)(m+1)}{N+1}\right]\sin\left[\frac{\pi(k+1)(n+1)}{N+1}\right] \tag{6.3.14}$$

和：

$$g(i,k)=\frac{2}{N+1}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}G_s(m,n)\sin\left[\frac{\pi(i+1)(m+1)}{N+1}\right]\sin\left[\frac{\pi(k+1)(n+1)}{N+1}\right] \tag{6.3.15}$$

DST 的酉核矩阵的元素为：

$$T_{i,k}=\sqrt{\frac{2}{N+1}}\sin\left[\frac{\pi(i+1)(k+1)}{N+1}\right] \tag{6.3.16}$$

不同于其他的正弦型变换，DST 对于 $N=2^p-1$ 计算最方便，其中 p 为整数。这样，它可以作为一个特地构造的（ $2N+2$ ）点的 FFT 的虚部来对待。

DST 有快速实现算法，它还具有一些适于某些图像压缩问题的性质。

4. 离散哈特利变换（Hartley）

1942 年，哈特利（Hartley）引入了一种连续积分变换作为傅立叶变换的代替。之后，Rracewell 基于哈特利变换定义了一种类似的离散酉变换。

二维离散哈特利正变换如下：

$$G_{m,n}=\frac{1}{N}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}g_{i,k}cos\left[\frac{2\pi}{N}(im+kn)\right] \tag{6.3.17}$$

6.3.2 方波型变换

有几种用于数字图像处理的变换，它们的基函数不是正弦型函数，而是方波的各种变形。通常这些变换的计算速度比较快，主要是因为许多乘法的操作都非常的简单。包括哈达玛（Hadamard）变换（1893 年）、沃尔什（Walsh）变换（1923 年）、斜（Slant）变换（1971

年）和哈尔（Haar）变换（1955年）。

特点：计算速度快（变换由整数组成，没有复数计算）。

1. 哈达玛变换（沃尔什变换）

$$T=\frac{1}{N}H_N X H_N \tag{6.3.18}$$

其中，$H_N=\begin{bmatrix}H_{N/2} & H_{N/2}\\ H_{N/2} & -H_{N/2}\end{bmatrix}$，$N=2^n$，$H_2=\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$。

例如：

$$H_4=\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & 1 & -1 & -1\\ 1 & -1 & -1 & 1\\ 1 & -1 & 1 & -1\end{bmatrix}\begin{matrix}0\\1\\2\\3\end{matrix} \tag{6.3.19}$$

其中，最后一列为列率，即每行符号变化的次数。对于每一行这个数都是不同的。这一符号的变化次数被称作这个行的“列率（Sequency）”。

基本图像的四条边为：

$$Wal(0)=[1\ \ 1\ \ 1\ \ 1]$$
$$Wal(1)=[1\ \ 1\ \ -1\ \ -1]$$
$$Wal(2)=[1\ \ -1\ \ -1\ \ 1]$$
$$Wal(3)=[1\ \ -1\ \ 1\ \ -1] \tag{6.3.20}$$

2. 斜（Slant）变换

（1）在图像压缩中，希望图像的能量集中在变换域少数几个变换系数上。

（2）若图形与某种变换的基本图像愈“相似”，则变换后的能量愈集中。

（3）通常，图像的灰度是逐渐改变的。

斜变换具有常数第一基函数和线性第二基函数（如图6-2所示）。倾斜的第二基函数可以用来匹配许多图像都有的线性升降的背景。斜变换的酉矩阵可以从2×2的哈尔或者哈达玛矩阵开始，通过下面的方式产生：

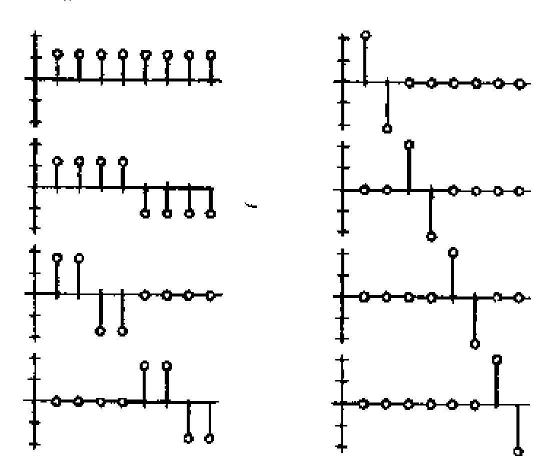


图6-2 N＝8时的斜变换基函数

$$S_2=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \tag{6.3.21}$$

$$S_N=\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&0\ 0&1&0&0\ 0\\a_N&b_N&0\ 0&-a_N&b_N&0\ 0\\0&0&&0&0&\\0&0&I&0&0&I\\0&1&0\ 0&0&-1&0\ 0\\-b_N&a_N&0\ 0&b_N&a_N&0\ 0\\0&0&&0&0&\\0&0&I&0&0&-I\end{bmatrix}\begin{bmatrix}S_{N/2}&0\\0&S_{N/2}\end{bmatrix} \tag{6.3.22}$$

其中 I 是阶数为 $N/2-2$ 的单位阵，且：

$$a_{2N}=\sqrt{\frac{3N^2}{4N^2-1}}\text{，且 }B_{2N}=\sqrt{\frac{N^2-1}{4N^2-1}} \tag{6.3.23}$$

3. 哈尔（Haar）变换（一种特殊的小波变换）

傅立叶变换的基函数间仅仅是频率的不同，哈尔函数在尺度（宽度）和位置上都不同。这是哈尔变换具有尺度和位置的双重属性，这在其基本函数（如图 6-3 所示）中十分明显。这样的属性使得它与前面提到的其他变换不太相同。

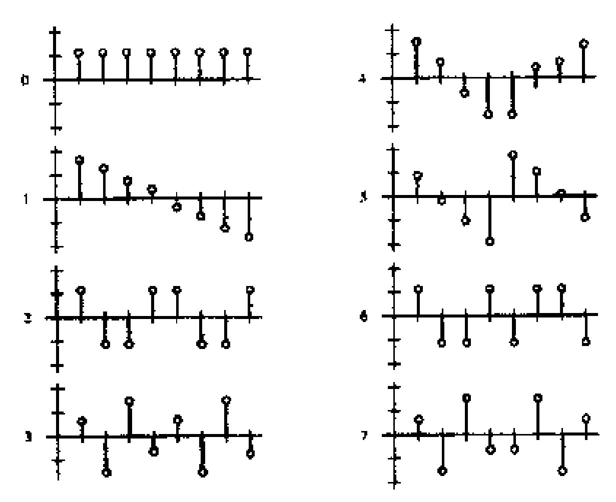


图 6-3　N = 8 时哈尔变换基函数

哈尔函数定义在[0，1]区间，令整数 $0\leqslant k\leqslant N-1$，由整数 q、p 惟一决定，即：

$$k=2^p+q-1 \tag{6.3.24}$$

其中，2^p 是使 $2^p\leqslant k$ 的 2 的最大幂，而 $q-1$ 为余数。

哈尔函数被定义为：

$$h_0=\frac{1}{\sqrt{N}} \tag{6.3.25}$$

且：

$$h_k(x)=\frac{1}{\sqrt{N}}\begin{cases}2^{p/2}\ , & \dfrac{q-1}{2^p}\leqslant x<\dfrac{q-1/2}{2^p}\\-2^{p/2}\ , & \dfrac{q-1/2}{2^p}\leqslant x<\dfrac{q}{2^p}\\0\ , & \text{其他}\end{cases} \tag{6.3.26}$$

其中，p 规定了尺度，q 决定了平移。

以前讨论过的各种变换都使用全宽度基函数，而哈尔函数则是对形式为一个矩形脉冲对的“原型”函数进行尺度变换和平移得到的。这个性质有两个主要的方面：

（1）虽然基函数可以由单一的索引 k 来决定，但它们都有由索引 p 和 q 规定的尺度、位置双重属性。这样，沿着 k 轴来画它的变换系数就不像传统的傅立叶变换得到的频谱那样可以给出更具启发性的信息了。

（2）假定在信号中沿着 X 轴的某一位置有一个特征，则傅立叶变换可以按照平移理论将这个位置编码到相位谱中。尽管这个特征的位置可以惟一地被确定，并通过傅立叶反变换而被完全恢复，但它在谱中并不能很直观地显示出来。

与之相反，哈尔变换直接地反映线和边，这是由于它的基函数有类似的特征。可以回忆一下，如果一个信号，或信号中的一部分可以近似地匹配上某一个基函数，则在变换后，会产生一个对应那个基函数的较大的变换系数。由于基函数是正交的，则这个信号对于其他的基函数将产生较小的系数。这样哈尔变换可以给出一些线和边的尺寸和位置的信息。

哈尔变换的 8×8 酉核心矩阵为：

$$Hr=\frac{1}{\sqrt{8}}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix} \tag{6.3.27}$$

对于更大的 N，也有同样的样式。由于矩阵中有许多常数和零值，哈尔变换可以非常快地计算出来。

6.4　基于特征向量的变换

通过对一类图像特征分析得出变换核函数。

6.4.1　特征分析

对于 $N\times N$ 矩阵，有 N 个常数 λ_k，$k=0,\dots,N-1$，使得：

$$\left|A-\lambda_k I\right|=0 \tag{6.4.1}$$

则称 λ_k 为矩阵 A 的特征值。

相应地，满足：

$$AV_k=\lambda_k V_k \tag{6.4.2}$$

式（6.4.2）的非零解 V_k 称为 A 的特征向量，V_k 为 $N\times 1$，每个 V_k 对应一个特征值 λ_k。如果 A 是对称的，则所有的特征向量构成一个正交基集。

6.4.2　主分量分析（PCA）

霍特林（Hotelling）提出可以去掉一个随机向量元素间相关性的线性变换（PCA）。卡胡南（Karhunen）和列夫（Loeve）对连续函数提出类似变换，并派生出离散图像变换的方

法，称为 K-L 变换。

L 个 N 维空间的向量，构成 N 维空间的 L 个点。如果大多数点落在一个 M 维超平面上，只要能找到 M 维空间的坐标系，则可以将 L 个向量投影到 M 维空间，获得低维的表达，如图 6-4 所示。

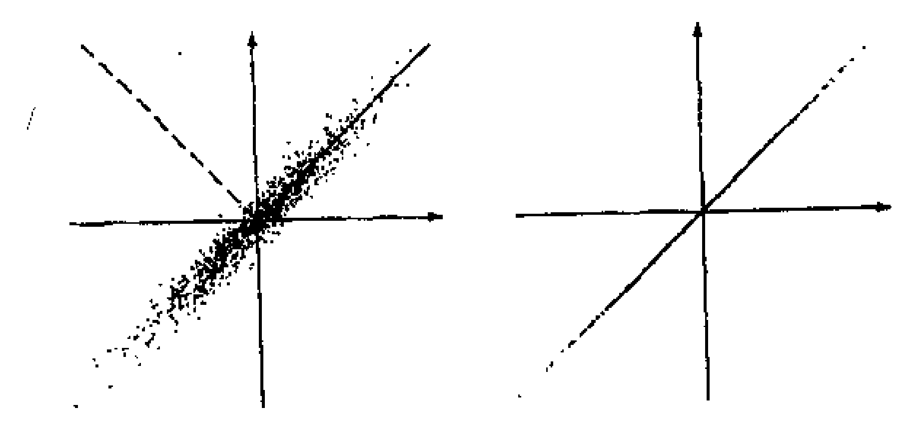


图 6-4 K-L 变换

设 x 是一个 $N\times1$ 的随机向量，x 的均值可以通过 L 个样本向量来估计：

$$\overline{x}\approx\frac{1}{L}\sum_{l=1}^{L}x_l \tag{6.4.3}$$

协方差矩阵为：

$$C_x=\varepsilon\left\{(x-\overline{x})(x-\overline{x})^t\right\}\approx\frac{1}{L}\sum_{l=1}^{L}x_lx_l^T-\overline{x}\overline{x}^T \tag{6.4.4}$$

其中，C_x 是半正定的实对称矩阵。

对 C_x 进行特征分析，可以获得一组实的非负特征值 $\lambda_1,\lambda_2,\ldots\lambda_r$ 及对应的特征向量 $V_1,V_2,\ldots V_r$，满足 $C_xV_i=\lambda_iV_i$。

令：

$$T=(V_1,V_2,\ldots,V_r)^T \tag{6.4.5}$$

则：

$$C_xT^T=[\lambda_i]T^T \tag{6.4.6}$$

作变换：

$$y=T(x-\overline{x}) \tag{6.4.7}$$

易证：

$$C_y=TC_xT^t=T[\lambda_i]T^T=[\lambda_i]=\begin{bmatrix}\lambda_1 & \cdots & 0\\ \cdots & \cdots & \cdots\\ 0 & \cdots & \lambda_N\end{bmatrix} \tag{6.4.8}$$

即 y 各元素互不相关。T 为相关变换。矩阵 λ_k 为变换后第 k 行变量 y_k 的方差。

由于公式（6.4.7）中的变换是可逆的，就是说可以通过变换向量 y 来重构 x。

$$x=T^{-1}y+\overline{x}=T^Ty+\overline{x} \tag{6.4.9}$$

降维：

将特征根从大到小排列，仍记为 $\lambda_1\geqslant\lambda_2\geqslant\lambda_3\geqslant\ldots\geqslant\lambda_r$。

同时将特征向量做相应排列，仍记为 $V_1,V_2,V_3,\ldots,V_r$。

取$1<M<r$，使：

$$\frac{\sum_{i=1}^{M}\lambda_i}{\sum_{i=1}^{r}\lambda_i}>\alpha \quad (6.4.10)$$

如$\alpha=95\%$。

令：

$$B=(V_1,V_2,\ldots,V_M)^t \quad (6.4.11)$$

$$\hat{y}=B(x-\bar{x}) \quad (6.4.12)$$

而x仍可通过$\hat{x}=B^t\hat{y}+\bar{x}$来（近似）重构。近似重构的均方误差为：

$$MSE=\sum_{k=M+1}^{r}\lambda_k \quad (6.4.13)$$

它只是与被舍弃的特征向量相对应的特征值的和。通常特征值幅度差别很大，所以忽略其中一些较小的值不会引起很大的误差。

6.4.3 卡胡南-列夫变换（K-L变换）的应用

公式（6.4.7）定义了一个（一维）离散变换、叫做卡胡南-列夫（K-L）变换、霍特林变换、特征向量变换和主分量法。K-L变换是压缩与特征提取的有效方法。

K-L变换的降维能力使其对于图像压缩非常有用。例如多光谱图像的每个像素有着多个灰度值，每个灰度值对应一个谱带。这样，一幅1000×1000的24通道多光谱图像可以被看作是一百万个24元随机向量（即像素）的集合。

例如多光谱图像：1000×1000点24通道多光谱图像。

（1）若取：

x_1　　$1000\times1000=1000^2\times1$

x_2　　$1000\times1000=1000^2\times1$

⋮

x_{24}　　$1000\times1000=1000^2\times1$

按公式（6.4.4）构造协方差矩阵，即：

$$X=[x_1-\bar{x},x_2-\bar{x},\ldots,x_{24}-\bar{x}]^t \quad (6.4.14)$$

$$C_x=\frac{1}{24}XX^t \quad (6.4.15)$$

其中，C_x为$1000^2\times1000^2$的矩阵，秩最多为24。

计算特征值和特征向量的计算量为$O(n^3)$，其中n为1000^2。

几乎不可能计算C'_x的特征值和特征向量。

（2）若取：

x_1　　24×1

x_2　　24×1

……

$x_{1000\times1000}$　　24×1

$$C_x' = \frac{1}{1000^2} X^T X \tag{6.4.16}$$

其中，C_x' 为 24×24 的矩阵，秩最多为 24。

通常，二维 K-L 变换的基图像取决于被变换的特定图像的统计特征，而不能写出显式形式。然而，如果这副图像是一阶马尔可夫过程，即像素间的相关性随像素间距离增大而线性递减，则可以用显式形式写出 K-L 变换的基图像。通常遇到的图像都能满足这个马尔可夫假设。

另外，相邻像素间的相关系数接近 1 时，K-L 变换的基函数接近余弦变换的基函数。于是，对于通常遇到的图像，都可以用更容易计算的离散余弦变换来近似 K-L 变换。

定理（SVD），设 A 是一秩为 r 的 $n\times r$ 维矩阵，则存在两个正交矩阵：

$$u = [u_0, u_1, \ldots u_{r-1}] \in R^{n\times r} \qquad u^T u = I \tag{6.4.17}$$

$$v = [v_0, v_1, \ldots v_{r-1}] \in R^{r\times r} \qquad v^T v = I \tag{6.4.18}$$

以及对角阵：

$$\Lambda = diag[\lambda_0, \lambda_1, \ldots \lambda_{r-1}] \in R^{r\times r} \tag{6.4.19}$$

$$\lambda_0 \geqslant \lambda_1 \geqslant \ldots \geqslant \lambda_{r-1} \tag{6.4.20}$$

满足：

$$A = u\Lambda^{1/2} v^T \tag{6.4.21}$$

其中，$\lambda_i (i = 0, 1, \ldots r-1)$ 为矩阵 AA^T 和 $A^T A$ 的非零特征值，u_i 和 v_i 分别为 AA^T 和 $A^T A$ 对应于 λ_i 的特征向量。

上述分解称为矩阵 A 的奇异分解（SVD），$\sqrt{\lambda_i}$ 为 A 的奇异值。

推论：

$$u = Av\Lambda^{-\frac{1}{2}} \tag{6.4.22}$$

即可取 AA^T 与 $A^T A$ 中维数较小的矩阵计算特征向量。

对于一组类似 A，使用同一 u，v^r 则可以有效压缩，因为只需保存特征值 Λ。

SVD 变换：对于特殊一点，任意一个 $N\times N$ 的矩阵 A 都可以写成：

$$A = U\Lambda V^T \tag{6.4.23}$$

其中 U 和 V 分别是 AA^T 和 $A^T A$ 的特征向量，Λ 为 $N\times N$ 的对角阵，沿其对角线包含 A 的奇异值。

由于 U 和 V 是正交的，所以：

$$\Lambda = U^T AV \tag{6.4.24}$$

公式（6.4.23）和（6.4.24）分别是一个酉变换的正变换和反变换。这个变换叫做奇异值分解（SVD）变换。如果 A 是对称的，则 $U = V$。

对于 Λ 是对角阵，所以它至多有 N 个非零元素。这样，可以至少获得 N 倍的无损压缩，如果 Λ 有一些为零（或很小）的奇异值，则可以获得更高的压缩比。这样，就可以通过进行一些额外的计算来获得很高的压缩比。

通常，奇异值中总有一些很小，小到可以忽略不计，而几乎不会带来什么误差。这样，可以通过忽略一些较小的 $\Lambda_{i,i}$ 值来实现有损压缩。这样做产生的均方差就是那些被忽略的奇异值的和。

SVD 变换表面上奇迹般的压缩能力使人感到迷惑。但实际上，虽然整幅图像可以被压缩成 Λ 的对角元，但对于每幅被压缩的图像来说，核矩阵 U 和 V 都是惟一的，这就是说，它们也必须被传输到接收方，才能重构原图像。然而，对于一组类似的图像来说，也可以只是近似地使用同一对核矩阵。

一个数字实例 SVD 变换如图 6-5 所示，使用的是一个对称的 5×5 的图像矩阵。

$$A=\begin{bmatrix}0&1&2&1&0\\1&3&4&3&1\\2&4&5&4&2\\1&3&4&3&1\\0&1&2&1&0\end{bmatrix},\quad AA^T=\begin{bmatrix}6&14&18&14&6\\14&36&48&36&14\\18&48&65&48&18\\14&36&48&36&14\\6&14&18&14&6\end{bmatrix},\quad \lambda=\begin{bmatrix}147.07\\1.872\\0.058\\0\\0\end{bmatrix}$$

$$U=\begin{bmatrix}0.186&0.638&0.241&-0.695&-0.695\\0.476&0.058&-0.52&-0.133&-0.128\\0.691&-0.422&0.587&0&0\\0.476&0.058&-0.52&0.133&0.128\\0.186&0.638&0.241&0.695&0.695\end{bmatrix}$$

$$\Lambda=U^tAU=\begin{bmatrix}12.585&0&0&0&0\\0&-1.142&0&0&0\\0&0&0.557&0&0\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix},\quad A=U\Lambda U^T=\begin{bmatrix}0&1&2&1&0\\1&3&4&3&1\\2&4&6&4&2\\1&3&4&3&1\\0&1&2&1&0\end{bmatrix}$$

正变换　　　　　　　　　　反变换

图 6-5　一个对称的 5×5 图像矩阵的 SVD 变换

K－L 变换在人脸识别上的应用：

（1）图像归一化。每张像用双眼对齐。

（2）采用直方图修正法使图像具有统一的均值和方差。

（3）使用部分人像做训练集计算特征根和特征向量。

例如：一张 $d\times d$ 人脸像，构成 d^2 维向量：

$$X=(b_{11}b_{12},\ldots,b_{bl}b_{21}b_{22},\ldots,b_{2d}b_{d1}b_{d2},\ldots,b_{dd})^T \tag{6.4.25}$$

一张人脸是 d^2 维空间一点，所有人脸都有五官，相似，样本值处于一个超平面（或人脸的像素间有很大的相关性）。

通过 K－L 变换找超平面的坐标基。

设有 N 类人像 $\{W_i, i=1,2,\ldots,N\}$，每类取 j 幅像构成训练样本集，其中有 M（$=N*j$）幅人脸 $\{x_i, i=0,1,\cdots M-1\}$，协方差矩阵为：

$$\sum=\varepsilon\{(\mathrm{X}-\overline{\mathrm{X}})(\mathrm{X}-\overline{\mathrm{X}})^T\} \tag{6.4.26}$$

或：

$$\sum=\frac{1}{M}\sum_{i=0}^{M-1}(x_i-\overline{\mathrm{X}})(x_i-\overline{\mathrm{X}})^t \tag{6.4.27}$$

其中：x_i 为第 i 个训练样本的图像向量：

$$\bar{X}=\frac{1}{M}\sum_{i=0}^{M-1}x_i \tag{6.4.28}$$

$\sum$ 是 $d^2\times d^2$ 维的对称矩阵，半正定的，进行特征分析：

$$\sum v=\lambda v \tag{6.4.29}$$

直接计算特征向量和特征根困难！计算复杂性为 $O((d^2)^3)$。

事实上：

$$\sum=\frac{1}{M}\sum_{i=0}^{M-1}(x_i-\bar{x})(x_i-\bar{x})^T=\frac{1}{M}XX^T,\quad d^2\times d^2 \tag{6.4.30}$$

其中：

$$X=[x_0-\bar{x},x_1-\bar{x},\cdots,x_{M-1}-\bar{x}] \tag{6.4.31}$$

故，构造矩阵：

$$R=X^TX\in R^{M\times M}\qquad M<<d^2 \tag{6.4.32}$$

求出特征根 λ_i 和对应的特征向量 $v_i(i=0,1,2,\cdots,M-1)$，由 SVD 定理：$\sum$ 的特征向量 u_i 为：

$$u_i=\frac{1}{\sqrt{\lambda_i}}Xv_i\qquad i=0,1,2,\cdots M-1 \tag{6.4.33}$$

1. 特征选择

令：

$$\lambda_0\geqslant\lambda_1\geqslant\cdots\geqslant\lambda_{r-1} \tag{6.4.34}$$

$u_0,u_1,\cdots,u_{r-1}$ 重排序，选择前 k 个特征向量，使：

$$\frac{\sum_{i=0}^{k-1}\lambda_i}{\sum_{i=0}^{M-1}\lambda_i}\geqslant\alpha,\quad \alpha=99\% \tag{6.4.35}$$

2. 人像表达

由 k 个特征向量做正交基，建立的子空间为像特征空间。

u_i 称为特征脸，记：

$$u=[u_0,u_1,u_2,\cdots,u_k] \tag{6.4.36}$$

对任意人像 f，向“特征脸”子空间投影，可获得投影的系数向量：

$$y=u^Tf \tag{6.4.37}$$

即为 f 的特征表达。

3. 识别测试

选择一个相似性度量 $d(x,y)$，例如欧氏距离：

$$d(x,y)=\sqrt{\sum_{i=1}^{N}(x_i-y_i)^2} \tag{6.4.38}$$

制作每个人的标准样本 A_i（$i=1,2,3,\cdots,N$）。

对 $\forall f$，求出 y，计算 y 与 A_i 的距离 $d(y,A_i)$。

若：

$$D(y,A_i)=\min_{j=1,2,\cdots N}D(y,A_j)\Rightarrow y\in W_i \tag{6.4.39}$$

6.5　变换域滤波

线性滤波（一个线性移不变系统）的作用，可以模型化为一幅图像的傅立叶谱乘以一个在频域（即变换）中定义的传递函数。尽管这个重要的结果仅对傅立叶变换是正确的，但是对于其他的变换也可以定义类似的运算。

和傅立叶变换类似，一般的酉变换都是将一幅图像分解成基图像的加权和。正变换的过程决定了加权系数，而反变换将这些分解出的基图像合成原图像。

变换域滤波就是在用反变换重构原图像之前对加权系数进行修改。对于线性滤波来说，变换是傅立叶变换，而修改过程则是通过将谱乘以一个传递函数来实现的，对于一般意义上的滤波，系数矩阵先被修改（通过乘或者其他的方法），然后反变换产生滤波后的图像。

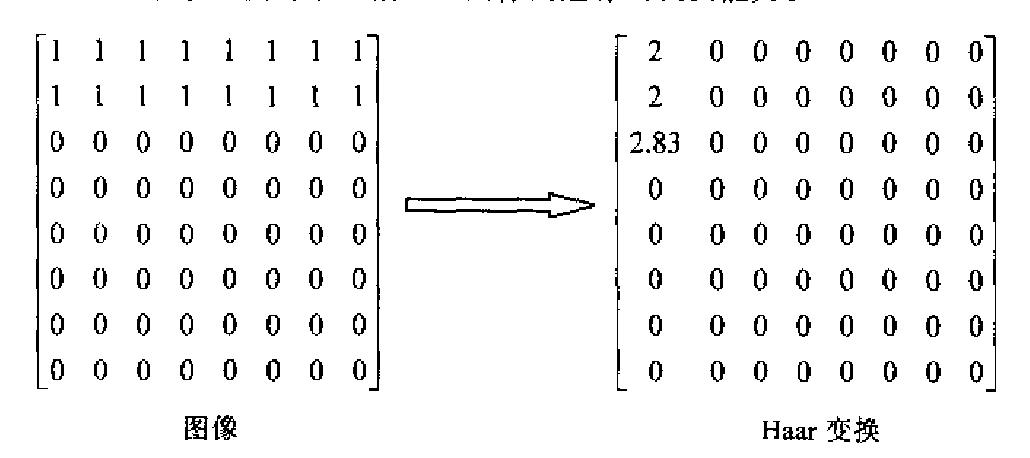
很明显，是基向量（及其产生的基本图像）的性质决定了各种变换的行为不同。例如，正弦型噪声在正弦型变换域中表现得非常紧凑，这可以很容易地通过将其相应的系数置零来消除。矩形波变换就不太适合于去除这种噪声，这是由于在变换域中噪声不能从信号中分离出来。

一般来说，如果图像中信号分量和噪声分量很像某个变换的某个和某几个基图像，则这个变换就可以用来分离信号和噪声。这是由于那些分量在频域中会以紧缩的形式表示。这个一般的结论对于去除噪声和信号检测都是适用的。

例如哈尔变换就是检测竖直和水平线以及边的很好选择，这是由于它的一些基图像与那些特征特别地匹配。

6.5.1　边、线和点的检测

图 6-6 显示了哈尔变换对于一幅 8×8 图像的边缘的检测能力。



$$\begin{bmatrix}1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix} \Longrightarrow \begin{bmatrix}2&0&0&0&0&0&0&0\\2&0&0&0&0&0&0&0\\2.83&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix}$$

图像　　　　Haar 变换

图 6-6　8×8 图像中的边缘检测

由于变换是可分离的，图像中竖直或水平的线或边缘特征仅分别在变换图像的第一行和第一列产生非零项。

在哈尔变换中，图 6-6 中的这种特征至多产生 $N/2$ 个非零项。特征的位置决定了哪个和多少个项非零。在其他变换中，第一行或列的所有 N 个项通常都非零。

图 6-7 中显示了一幅包含一个单像素冲激（尖峰）的图像的几种变换。除了哈尔变换以外（仅有 $2N$ 个非零项），其他的变换的所有 N^2 个元素都非零。同样，非零项的位置由

冲激的位置决定。

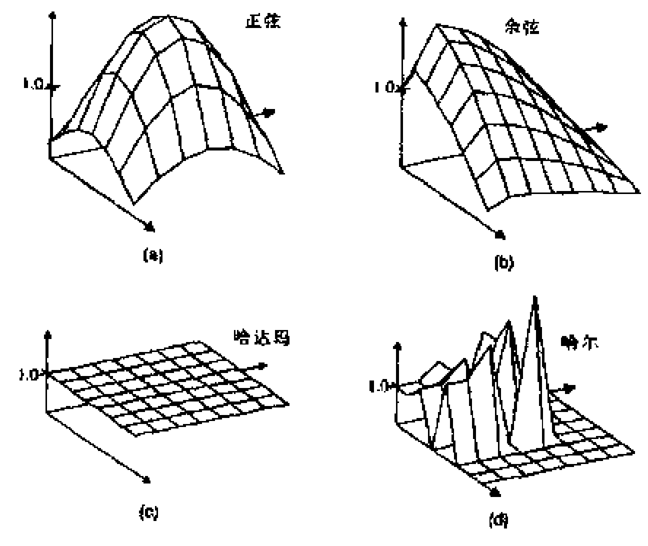


图 6-7　包含一个冲激的图像变换

（a）DST。

（b）DCT。

（c）哈达玛。

（d）哈尔。

输入的是一个 8×8 的矩阵，除了左上角元素为 8 以外，其他元素都为 0。

6.5.2　滤波器设计

由于傅立叶变换与线性移不变系统的密切联系，使得它有一个很好的理论背景来指导它在图像滤波中的应用。其他的变换的理论支持较少，所以它们的应用侧重于经验性的。理解这些变换间的异同，可以帮助指导读者寻找更好的解决问题的方法。

小结

本章主要介绍了线性变换、基函数和基图像、正弦型变换和方波型变换、基于特征向量的变换、变换域滤波等内容。

重点：一个 $N \times N$ 变换矩阵的行是 N 维向量空间的一组正交基向量。线性酉变换产生一个含有 N 个变换系数的向量，每个变换系数是输入向量与变换矩阵的某行的内积。类似的，反变换是通过变换系数向量与反变换矩阵的行的内积生成的。反变换也可以看作是构造基向量的一个加权和，变换系数就是权。对于一个二维对称可分离的酉变换，基图像是变换矩阵不同行之间的外积。

难点：理解主向量分析（PCA）和 K-L 变换的原理和应用。

疑点：在将数字图像进行处理时，是否需要特别考虑它对应的连续图像是什么样子。

练习六

一、选择题

1. 对于一个可分离的酉变换，每幅基本图像就是变换矩阵某两行的（　　）。

A. 和　　B. 差　　C. 内积　　D. 外积

2. 对于一个可分离的酉变换，每幅基本图像就是变换矩阵某两行的（　　）。

A. 乘积　　B. 内积　　C. 外积　　D. 各项之和

3. 基图像被认为是分解原图像所得的单位集分量，它们同时也是组成原图像的基本结构单元。正变换通过（　　）来实现分解，反变换通过将基图像（　　）来实现重构。

A. 确定系数　　B. 分解系数　　C. 加权求和　　D. 加权外积

二、思考题

1. 基图像对于数字图像处理有什么重要的意义。
2. K-L 变换对图像压缩有什么意义。

三、上机题

1. 用 MATLAB 编写一个程序，将一幅图像进行一维余弦变换 dct(X,n)，n 为周期，周期分别取 n = 1, n = 2, n = 5, n = 100。

2. 编写一个程序，将一幅图像进行二维余弦变换和反变换。

第 7 章　滤波器设计

在第 3 章和第 4 章中，介绍了分析和设计线性滤波器的基础知识。本章将讨论如何设计能实现特定目标的滤波器。为了便于理解，首先研究某些简单但是十分有用的滤波器的时域和频域的特征。在本章的后半部分，讨论为特定任务设计最优滤波器的问题。

表征线性系统特征的传递函数 $G(s)$ 或冲激响应 $g(t)$，都称为滤波器。滤波器设计本质上是线性系统的设计。滤波器有时域和频域两种形式，它们是一对傅立叶变换。正如第 3 章和第 4 章中一样，从分析一维（时间变量 t）信号入手，这样做既简化了数学运算又便于图示。而从一维推广到二维是很直截了当的。

本章主要内容如下：

（1）低通滤波器。

（2）带通或带阻滤波器。

（3）高频增强滤波器。

（4）最优线性滤波器设计。

（5）排序统计滤波器。

7.1　低通滤波器

信号或图像的能量集中在幅度谱的低频或中频段是常见的，而在较高频段，感兴趣的信息常被噪声所淹没；所以需要一个能降低高频成分幅度的滤波器，用以减弱噪声的影响。

低通滤波器就是能量集中在幅度谱的低频部分的滤波器（如图 7-1 所示）。

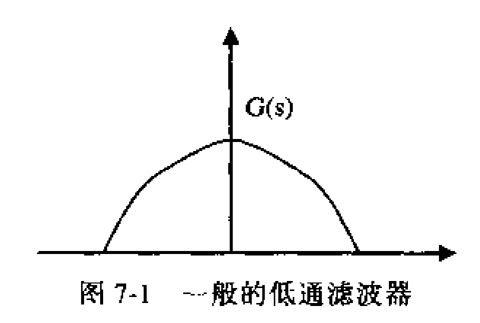


图 7-1　一般的低通滤波器

低通滤波器的作用就是要滤掉高频成份，仅保留低频或中频成份（如图 7-2 所示）。

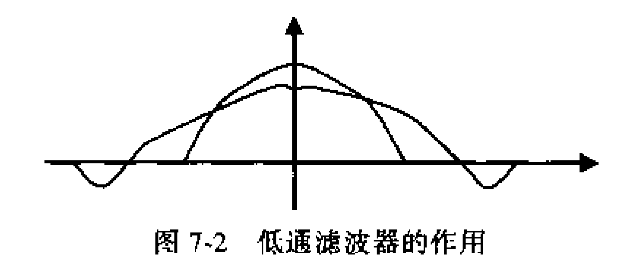


图 7-2　低通滤波器的作用

$$H(s) = F(s) \bullet G(s) \qquad (7.1.1)$$

7.1.1　常用低通滤波器

下面是一些常用的低通滤波器和对应的传递函数，如图 7-3 所示，任何一个信号 $f(t)$ 与

滤波器 $g(t)$ 满足如下的关系式：

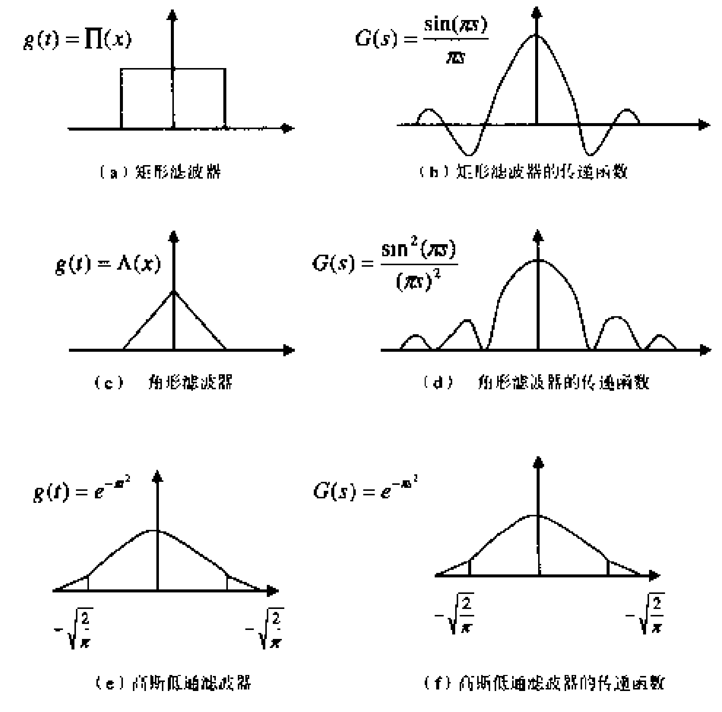


图 7-3　常用的低通滤波器

（a）（c）（e）分别是矩形滤波器、三角形滤波器、高斯低通滤波器。

（b）（d）（f）分别是滤波器所对应的传递函数（频域）。

$$F\{f(t)*g(t)\}=F(s)\bullet G(s) \tag{7.1.2}$$

7.1.2　空间域

$f(t)$ 与 $\Pi(t)$（矩形脉冲），$\Lambda(t)$（三角脉冲）或 $e^{-\pi t^2}$（高斯）卷积，效果即是将 $f(t)$ 局部平均、平滑（如图 7-4 所示）。

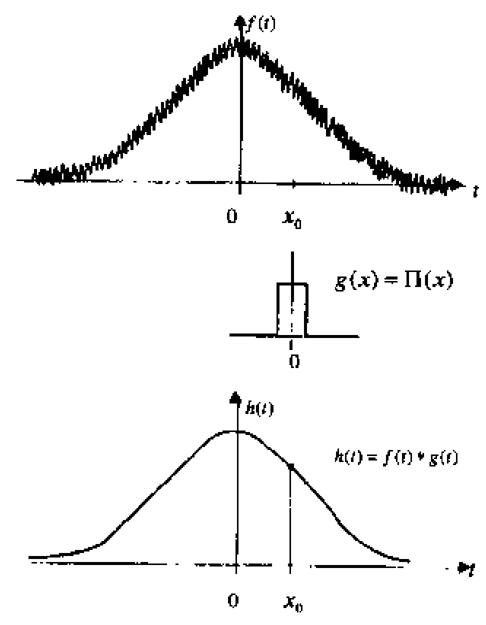


图 7-4　对有噪声的函数进行平滑

卷积得到 $h(t)$ 的运算如下：

$$h(t)=\int_{-\infty}^{\infty}g(t-\tau)f(\tau)d\tau \tag{7.1.3}$$

7.1.3 频域

矩形函数谱（如图 7-5 所示）：

$$G(s)=\frac{\sin(\pi s)}{\pi s} \tag{7.1.4}$$

注意：由相似性定理，传递函数的宽度与冲激响应宽度成反比。当冲激响应宽度太大，输出可能会改变极性。

三角函数谱（如图 7-6 所示）：

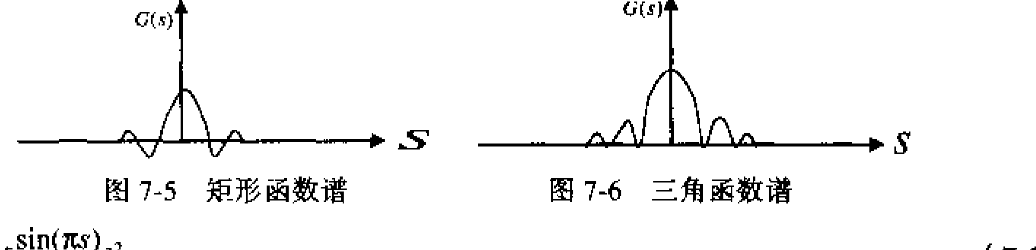


图 7-5 矩形函数谱　　图 7-6 三角函数谱

$$G(s)=[\frac{\sin(\pi s)}{\pi s}]^2 \tag{7.1.5}$$

注意：三角函数谱不会为负；且衰减速度大于矩形滤波。故在较大的宽度安全地使用不会发生极性反转。

7.1.4 高频截止

$$H(s)=F(s)\bullet\prod(s) \tag{7.1.6}$$

即：

$$h(t)=f(t)*\frac{\sin(\pi t)}{\pi t} \tag{7.1.7}$$

在尖峰或边界会产生振铃效应。

7.2 带通或带阻滤波器

信号中有用成分和希望除掉的成分分别出现在频谱的不同频段，需要设计允许或阻止特定频段通过的传递函数。

7.2.1 理想的带通滤波器

通过卷积，实现一个滤波器，它仅允许位于 f_1 和 f_2 之间 $(f_1<f_2)$ 的能量通过，即传递函数为：

$$G(s)=\begin{cases}1 & f_1<|s|<f_2\\ 0 & \text{其他}\end{cases} \tag{7.2.1}$$

若令：

$$s_0=\frac{1}{2}(f_1+f_2) \quad \text{和} \quad \Delta s=f_2-f_1\text{，则：} \tag{7.2.2}$$

$G(s)$ 可重写为：

$$G(s)=\prod(\frac{s-s_0}{\Delta s})+\prod(\frac{s+s_0}{\Delta s}) \\ =\prod(\frac{s}{\Delta s})*[\delta(s-s_0)+\delta(s+s_0)] \quad (7.2.3)$$

于是：

$$g(t)=\Delta s\frac{\sin(\pi\Delta st)}{\pi\Delta st}2\cos(2\pi s_0 t) \\ =2\Delta s\frac{\sin(\pi\Delta st)}{\pi\Delta st}\cos(2\pi s_0 t) \quad (7.2.4)$$

如图 7-7 所示，被频率为 $\frac{\Delta s}{2}$ 的包络线 $\frac{\sin(x)}{x}$ 所包围的，以 s_0 为频率的余弦波。当 $\Delta s\to 0$ 时，$G(s)$ 为一对脉冲，$g(t)$ 为频率为 s_0 的余弦信号。

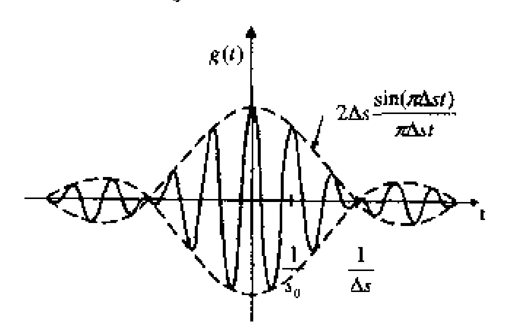


图 7-7 理想带通滤波器的传递函数

7.2.2 理想带阻滤波器

阻止频率位于 f_1 和 f_2 之间的能量通过，而允许其他频率的能量通过的滤波器。传递函数可表示如下：

$$G(s)=\begin{cases}0 & f_1\leqslant |s|\leqslant f_2 \\ 1 & \text{其他}\end{cases} \quad (7.2.5)$$

该传递函数如图 7-8 所示。

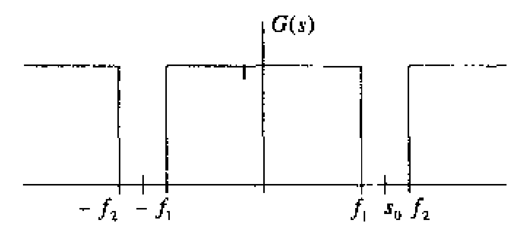


图 7-8 理想带通滤波器的传递函数

为了方便起见，仍然令 s_0 为中心频率，而 Δs 表示被阻频带的宽度；于是可以把传递函数重写为：

$$G(s)=1-\prod(\frac{s}{\Delta s})*[\delta(s-s_0)+\delta(s+s_0)] \quad (7.2.6)$$

于是：

$$g(t)=\delta(t)-2\Delta s\frac{\sin(\pi\Delta st)}{\pi\Delta st}\cos(2\pi s_0 t) \quad (7.2.7)$$

该冲激响应如图 7-9 所示。

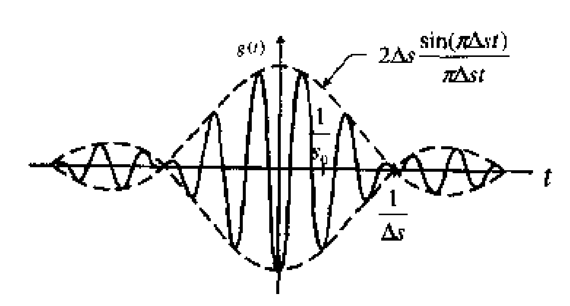


图 7-9 理想带阻滤波器的冲激响应

当 Δs 很小时，称这样的带阻滤波器为“切口”滤波器。

7.2.3 通用带通滤波器

由式（7.2.3），更一般地设 $K(s)$ 为一非负单峰函数，则可构造带通滤波器的传递函数，如图 7-10 所示。此传递函数用公式表示为：

$$G(s)=K(s)*[\delta(s-s_0)+\delta(s+s_0)] \tag{7.2.8}$$

其冲激相应为：

$$g(t)=2k(t)\cos(2\pi s_0 t) \tag{7.2.9}$$

其中：

$$k(t)=F^{-1}\{K(s)\} \tag{7.2.10}$$

其图为在 $k(t)$ 包络线内，频率为 s_0 的余弦信号。

例如：如图 7-11 所示，假设 $K(s)$ 是高斯函数，则：

$$G(s)=Ae^{-s^2/2\alpha^2}*[\delta(s-s_0)+\delta(s+s_0)] \tag{7.2.11}$$

那么冲激响应为：

$$g(t)=\frac{2A}{\sqrt{2\pi\sigma^2}}e^{-t^2/2\sigma^2}\cos(2\pi s_0 t) \tag{7.2.12}$$

其中：

$$\sigma=\frac{1}{2\pi\alpha} \tag{7.2.13}$$

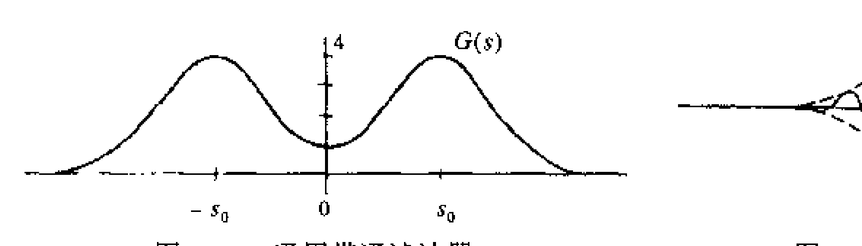


图 7-10 通用带通滤波器

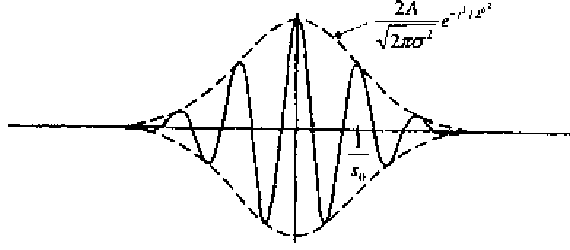


图 7-11 高斯带阻滤波器

7.3 高频增强滤波器

高频增强滤波器又称为高通滤波器，如图 7-12 所示，其传递函数满足：

（1）它的频值在 0 频率点为 1。

（2）随着频率的增长，传递函数的值逐渐增加。

（3）当频率增加到一定值以后，传递函数又回到0值或降低某个大于1值。

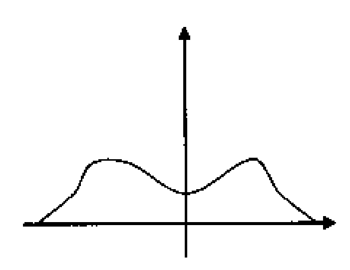


图7-12 高频增强滤波器

高频增强滤波器实际上类似一种带通滤波器；只不过规定0频率处的增益为1。高频增强滤波器不改变图像中面积较大的且缓慢变化的成分的对比度；在零点附近频率较低部分是图像中面积较大而且改变缓慢的地方，通过滤波器后，由于：

$$G(0)=1 \tag{7.3.1}$$

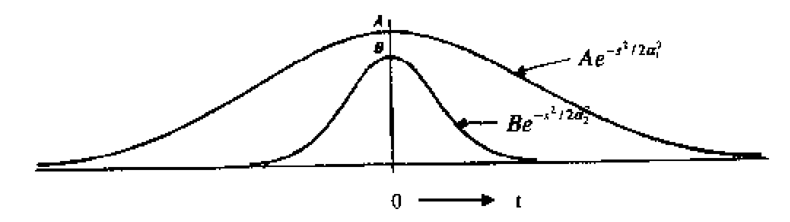
所以得到的相对应的输出为：

$$H(0)=F(0)G(0)=F(0) \tag{7.3.2}$$

即零频率处不改变对比度。实际应用中，为了减少图像面积大且缓慢变化的成分的对比度，有时让零频率处的增益小于单位1更合适。如果传递函数通过原点，则可以称为Lapalacian滤波器。

7.3.1 高斯差分（DOG）滤波器

用两个不同宽度的高斯函数之差分来产生高频增强的传递函数（如图7-13所示）。



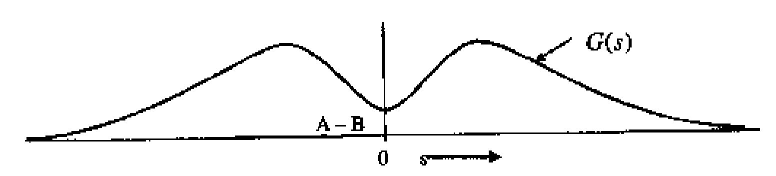


图7-13 高斯高频增强滤波器传递函数

$$G(s)=Ae^{-s^2/2\alpha_1^2}-Be^{-s^2/2\alpha_2^2} \qquad A>B，\alpha_1>\alpha_2$$

其冲激响应为：

$$g(t)=\frac{A}{\sqrt{2\pi\sigma_1^2}}e^{-t^2/2\sigma_1^2}-\frac{B}{\sqrt{2\pi\sigma_2^2}}e^{-t^2/2\sigma_2^2}$$

其中：

$$\sigma_i=\frac{1}{2\pi\alpha_i}$$

由相似性定理，频域较宽的高斯函数在时域产生较窄的高斯波形，反过来也如此。

当 $\alpha_1 \to \infty$，时域中窄高斯信号 $\to \delta(t)$。

如图 7-14 所示，频域较宽的高斯函数在时域产生较窄的高斯波形，反过来也如此。图 7-14 所示是带通滤波器的典型冲激响应：在一个负的源盘中有一个正的脉冲。

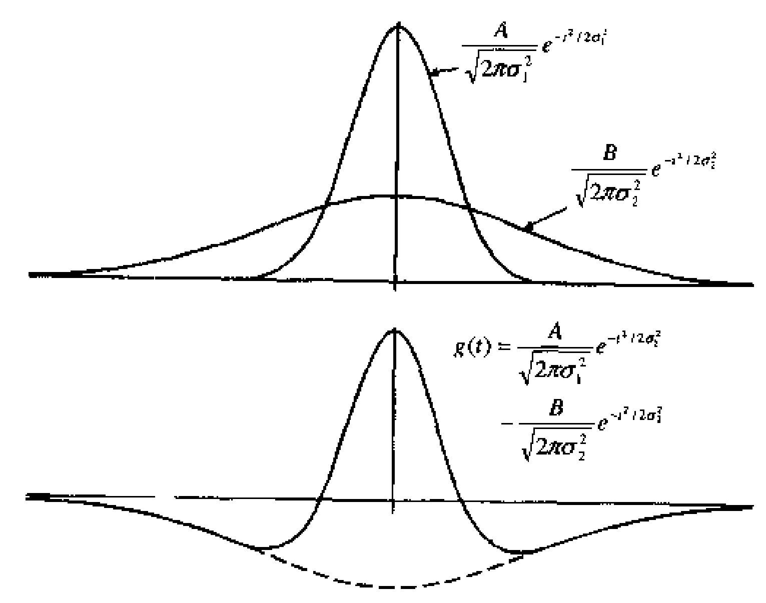


图 7-14 高斯高频增强滤波器的冲激响应

如果让 $\alpha_1 \to \infty$，那么时域的窄高斯信号将进一步变窄以至于变成冲激信号，这时的滤波器如图 7-15 所示。

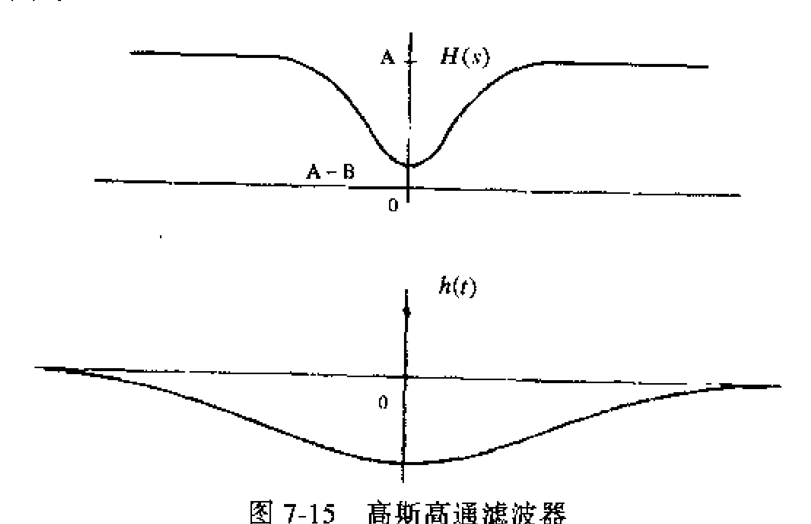


图 7-15 高斯高通滤波器

注意：在其传递函数的值在高频时降为 0 的滤波器与不降为 0 的滤波器之间的差正好是时域里中央脉冲的宽度。

事实上中央脉冲越宽，传递函数越早降到 0 值。

7.3.2 设计高通滤波器的经验规则

假定滤波器的冲激响应为一个窄脉冲减去一个宽脉冲：

$$g(t)=g_1(t)-g_2(t) \tag{7.3.3}$$

如图 7-16 所示，其传递函数 $G(s)$ 将具有通用的高频增强滤波器的形状。下面通过估计传递函数在 0 频率处的值来确定它对于图像中大目标对比度的影响。同时也估计传递函

数在任何频率处的最大值。

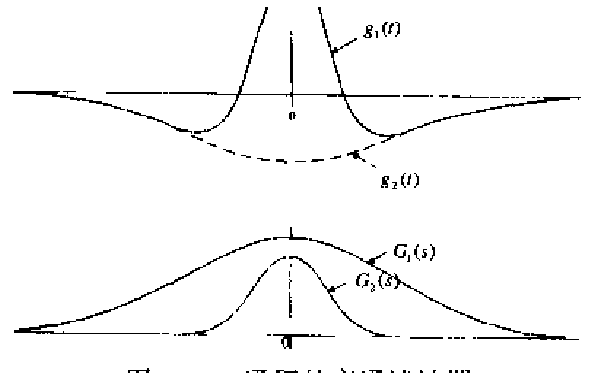


图 7-16　通用的高通滤波器

下面估计高频增强滤波器的两个经验规则：

（1）估计传递函数在0频率的值，确定对图像中大目标的对比度影响。

$$G(s)=\int_{-\infty}^{\infty} g(t)e^{-j2\pi st}dt \tag{7.3.4}$$

$$G(0)=\int_{-\infty}^{\infty} g(t)dt=\int_{-\infty}^{\infty} g_1(t)dt-\int_{-\infty}^{\infty} g_2(t)dt=A_1-A_2 \tag{7.3.5}$$

其中，A_1 和 A_2 分别代表 $g_1(t)$ 和 $g_2(t)$ 的面积。

（2）高频增强滤波器的最大值。

因为：$G=G_1-G_2$，$G_2\geqslant 0$

所以：$G_{max}\leqslant G_{1max}\leqslant G_1(0)=\int_{-\infty}^{\infty} g_1(t)dt=A_1$

对于两个脉冲的差构成的高频增强滤波器的两个经验规则为：

$G(0)=A_1-A_2$，$G_{max}\leqslant A_1$

低频响应：

讨论滤波器对图像中较大物体和恒值灰度区域的响应。

假定 $g(x)$ 持续时间有限的函数，$f(x)=c$。

$$h(x)=\int_{-\infty}^{\infty} f(\tau)g(x-\tau)d\tau=\int_{-\infty}^{\infty} cg(x-\tau)d\tau=c\int_{-\infty}^{\infty} g(x-\tau)d\tau=c\int_{-\infty}^{\infty} g(\tau)d\tau \tag{7.3.6}$$

由于：

$$G(0)=\int_{-\infty}^{\infty} g(x)dx \tag{7.3.7}$$

于是：

$$h(x)=cG(0) \tag{7.3.8}$$

因此，如果 $G(0)=1$，那么滤波器不改变 $f(x)$ 中大而且恒定区域的幅值（对比度）；若 $G(0)\neq 1$，则为增益因子，控制着 $h(t)$ 和 $f(t)$ 的较大分量之间的总体幅度关系。

7.4　最优线性滤波器设计

下面介绍在某种意义上对完成特定任务而言是最优的滤波器的设计技术。首先建立一套性能指标，然后通过选择合适的冲激响应（或传递函数）来最大化这些性能指标。

在数字图像处理的历史中，曾经出现非常多的滤波器设计，正如在第一次世界大战中的飞行技术一样是靠经验的。那时人们是从计算简单、以往的成功、方便性、审美要求、传闻、一时的想法等方面来选择滤波器的。这种滤波器设计可能是成功的，但是它带着非常不利的印记——只是次优的。它几乎从来就没有设计出最优的滤波器，并且可能是十分危险的。

次优滤波器——特别是那些易于用计算机实现的滤波器可能会给图像引入人为的现象而且常常是不可预料的。例如程序员非常喜爱的、在某个域是矩形的滤波器，由于函数无限起伏的原因，会在对应的域中产生令人厌恶的效果。

当人们在一个域里使用方形滤波器时，通常受在另一个域里出现的减幅振荡（振铃现象）和其他的人工效应所困扰。他们有时错误地认为这些缺点是数字信号处理所固有的，或者叹息计算机缺乏正确地完成工作的能力。

在本节中，将详细介绍最优滤波器设计技术，并且说明在一般情况下，这些最优滤波器工作得很好。掌握了这些知识后，就能在最优性和计算复杂性之间进行权衡取舍，而不会招致灾难性的人工效应。

先给出随机变量的概念，然后讨论两种最优滤波器的设计技术：

（1）维纳滤波器（Wiener filter）。



求 $h(t)$，使得 $y(t)=s(t)$。

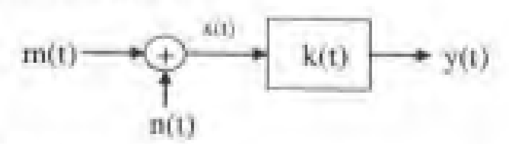
维纳滤波器的实例如图 7-17 所示。



图 7-17 维纳滤波器实例

使得 $MSE=\varepsilon\left\{e^2(t)\right\}=\int_{-\infty}^{\infty}e^2(t)dt$ 最小。其中，$e(t)=s(t)-y(t)$。

（2）匹配检测器（Matched Detector）。



求 $k(t)$，使得从 $y(t)$ 能找出 $m(t)$。

维纳滤波对于从加性噪声中恢复未知信号来说是最优的。而匹配检测器对于检测淹没在加性噪声中的已知信号来说是最优的。

7.4.1 随机变量

1. 随机变量

考虑一个无限多个函数成员构成的样本集 $A=\{n_i(t)\}_{i\in z}$，当记录信号时，其中的一个成员函数出现并污染了信号，但是无法知道是哪一个成员函数，然而，可以对样本集整体作一些总体描述，用这种方法来表示对造成污染的噪声信号的“部分”知识。

2. 遍历性随机变量

时间均值：$A=\int_{-\infty}^{\infty}x(t)dt$

样本均值：给定 $t=t_0$，$\varepsilon\{x(t)\}\left(=\frac{1}{M}\sum_{i=1}^{M}x_i(t)\right)$

定义：一个随机变量是遍历性的，当且仅当：所有成员函数的时间均值是相等的；样本集均值不随时间而改变；时间均值和样本集均值在数量上是相等的。

$$\varepsilon\{x(t)\}=\int_{-\infty}^{\infty}x(t)dt$$

现假设，$n(t)$，$s(t)$，$n(t)n(t+\tau)$，$e^2(t)$ 都为遍历的。且知道 $n(t)$，$s(t)$ 功率谱的知识。

$$R_n(\tau)=\int_{-\infty}^{\infty}n(t)n(t+\tau)dt$$

$$R_s(\tau)=\int_{-\infty}^{\infty}s(t)s(t+\tau)dt$$

$$P_n(s)=F\{R_n(\tau)\}=|N(s)|^2$$

$$P_s(s)=F\{R_s(\tau)\}=|S(s)|^2$$

这意味着知道 $n(t)$ 的幅度谱，但是不知道它的相位谱。实际上，成员函数集是由无限多个仅在相位谱上有区别的函数均值构成的。任何非负的实偶函数都可能是随机变量的功率谱，并且任何有非负谱的实偶函数都可以作为随机变量的自相关函数。

幸运的是，通常遇到的随机信号大都符合遍历性随机变量模型。例如，重复地观察“白噪声”源就可以发现，测得的频谱近似等于常数，不因频率而改变。

7.4.2 维纳滤波器

维纳滤波器是一种经典的线性降噪滤波器。尽管有能导出该滤波器主要结果的比较简单的方法，但是下面的讨论可解释维纳滤波器技术的全部能力。为简化起见，下面的推导过程是针对一维信号的。

假定所观察的信号 $x(t)$ 由真实信号 $s(t)$（有用的信号）和加性噪声信号 $n(t)$ 组成。如果设计一个滤波器，使得其在尽可能地降低噪声信号的同时恢复真实信号，就能用滤波器尽量接近地估计出原来的信号。

最理想的情况是让 $y(t)$（滤波器输出）等于 $s(t)$，但是一般情况下，线性滤波器的能力还不足以精确地恢复被噪声污染的信号。要做的是选择 $h(t)$（冲激响应）使得 $y(t)$ 尽可能地逼近 $s(t)$。

在下面的分析中，假设 $s(t)$ 和 $n(t)$ 都是遍历性随机变量，因而它们的功率谱是已知的。这意味着尽管不能确切地知道 $n(t)$，但是可以知道 $n(t)$ 来自某函数样本集，该集的所有成员函数有相同的自相关函数，因此有相同的功率谱。$s(t)$ 也做这样的限制。

1. 最优准则

在开始设计最优滤波器之前，必须建立最优性的客观标准。既然一般说来，要求线性滤波器能使 $y(t)=s(t)$ 不太实际，所以要求在此情况下能做得更好。下面将选用均方误差作为最优准则。

不管 $h(t)$ 是否是最优的，$s(t)$ 对于任何输入，都能得到相应的输出 $y(t)$。下面将滤波器输出端的误差信号定义为：

$$e(t)=s(t)-y(t) \tag{7.4.1}$$

即实际输出与所希望得到的输出的差，它是时间 t 的函数。以平均计，如果冲激响应 $h(t)$ 选择得合适，那么误差信号就会相当小。反过来，选择越差的 $h(t)$，误差就越大。所以选择 $h(t)$，使误差信号均方误差 $e(t)=s(t)-y(t)$ 最小。

用均方差作为平均误差的度量：

$$MSE=\varepsilon\left\{e^2(t)\right\}=\int_{-\infty}^{\infty}e^2(t)dt \tag{7.4.2}$$

其中，$e^2(t)$ 为遍历性随机变量。后面的等式成立是因为误差信号是遍历性随机变量 $e^2(t)$ 的组合，因而它本身也是随机变量。

注意：对于正的和负的误差来说，$e^2(t)$ 都是正值，而且对误差进行平方运算使得大误差的分量比小误差中的多。正是由于这些原因，这里才选取最小化均方误差作为最优准则。从直觉来说这是令人满意的选择。尽管还可能用到其他的准则（例如平均绝对误差），但是它们可能会使分析复杂化，而对目标来说只有很小的或者根本没有什么优越性。

2. 均方误差

现表述问题如下：给定 $s(t)$ 和 $n(t)$ 的功率谱，必须确定使均方误差最小的冲激响应。

注意：均方误差使冲击响应 $h(t)$ 的函数，它将一个函数 $h(t)$ 映射为一个实数 MSE。

关于函数最小化的数学分支是变分（Calculus of Variations），在这里要用到它。具体的步骤如下：

（1）得出用 $h(t)$ 表示的 MSE 的函数表达式。

（2）找出用已知的功率谱表示的最优冲激响应 $h_0(t)$ 的表达式。

（3）找出使用 $h_0(t)$ 时得到的 MSE 表达式。

由（7.4.1）式得：

$$\begin{aligned}MSE&=\varepsilon\left\{e^2(t)\right\}=\varepsilon\left\{\left[s(t)-y(t)\right]^2\right\}\\&=\varepsilon\left\{s^2(t)-2s(t)y(t)+y^2(t)\right\}\\&=\varepsilon\left\{s^2(t)\right\}-2\varepsilon\left\{s(t)y(t)\right\}+\varepsilon\left\{y^2(t)\right\}\end{aligned} \tag{7.4.3}$$

因为期望是一个积分算子，于是：

$$MSE=\varepsilon\left\{s^2(t)\right\}-2\varepsilon\left\{s(t)y(t)\right\}+\varepsilon\left\{y^2(t)\right\}=T_1+T_2+T_3 \tag{7.4.4}$$

引入 T_1、T_2 和 T_3 是为了能够独立地考虑这三项。将 T_1 写成积分形式，有：

$$T_1=\varepsilon\left\{s^2(t)\right\}=\int_{-\infty}^{\infty}s^2(t)dt=R_s(0) \tag{7.4.5}$$

这正是 $s(t)$ 的自相关函数在 $\tau=0$ 点的值，是已知的，所以 T_1 的值一开始就是已知的。

把 $y(t)$ 写为 $x(t)$ 和 $h(t)$ 的卷积则允许把第二项展开为：

$$T_2 = -2\varepsilon\left\{s(t)\int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau\right\} \tag{7.4.6}$$

因为期望算子实际上是对时间的积分，可以改变式（7.4.6）的积分次序得到：

$$T_2 = -2\int_{-\infty}^{\infty} h(\tau)\varepsilon\{s(t)x(t-\tau)\}d\tau \tag{7.4.7}$$

积分里的期望是 $s(t)$ 和 $x(t)$ 的自相关函数，于是有：

$$T_2 = -2\int_{-\infty}^{\infty} h(\tau)R_{xs}(\tau)d\tau \tag{7.4.8}$$

可以把 T_3 展开为两个卷积之积的期望：

$$T_3 = \varepsilon\left\{\int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau\int_{-\infty}^{\infty} h(u)x(t-u)du\right\} \tag{7.4.9}$$

和前面一样，经整理得：

$$T_3 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\tau)h(u)\varepsilon\{x(t-\tau)x(t-u)\}d\tau du \tag{7.4.10}$$

令 $v = t-u$，则：

$$\varepsilon\{x(t-\tau)x(t-u)\} = \varepsilon\{x(v+u-\tau)x(v)\} \tag{7.4.11}$$

它是 $x(t)$ 的自相关函数在 $u-\tau$ 处的取值。现在可以把第三项写为：

$$T_3 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\tau)h(u)R_x(u-\tau)d\tau du \tag{7.4.12}$$

于是，将均方误差写为：

$$MSE = R_s(0) - 2\int_{-\infty}^{\infty} h(\tau)R_{xs}(\tau)d\tau + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\tau)h(u)R_x(u-\tau)d\tau \tag{7.4.13}$$

这就是用滤波器得冲激响应以及已知的两输入分量的自相关和互相关函数表示的均方误差。正如所期望的，MSE 是 $h(t)$ 的函数。现在希望选取特定的函数 $h_0(t)$，使得 MSE 取得最小值。

3. 最小化 *MSE*

设 $h_0(t)$，使 MSE 达到最小，最小均方误差记为 MSE_0。一般来说，任意的 $h(t)$ 与最优的 $h_0(t)$ 之间是不同的，下面定义一个函数 $g(t)$ 来表示这个差异，即：

$$h(t) = h_0(t) + g(t) \tag{7.4.14}$$

这里 $h(t)$ 是一个任意选取的次优冲激响应函数，而 $g(t)$ 是被选取来使得等式成立的函数。这样看似乎表面上被复杂化了，但是它可以为 $h_0(t)$ 确立必要条件。

将（7.4.14）式代入（7.4.13）式，得：

$$\begin{aligned} MSE = R_s(0) &- 2\int_{-\infty}^{\infty}[h_0(\tau)+g(\tau)]R_{xs}(\tau)d\tau \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[h_0(\tau)+g(\tau)]\ [h_0(u)+g(u)]R_x(u-\tau)d\tau du \end{aligned} \tag{7.4.15}$$

展开得：

$$\begin{aligned} MSE = R_s(0) &- 2\int_{-\infty}^{\infty} h_0(\tau)R_{xs}(\tau)d\tau + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_0(\tau)h_0(u)R_x(u-\tau)d\tau du \\ &+ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_0(\tau)g(u)R_x(u-\tau)d\tau du + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_0(u)g(\tau)R_x(u-\tau)d\tau du \\ &- 2\int_{-\infty}^{\infty} g(\tau)R_{xs}(\tau)d\tau + \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(\tau)g(u)R_x(u-\tau)d\tau du \end{aligned} \tag{7.4.16}$$

比较式（7.4.16）的前三项与式（7.4.13）的前三项，会发现这三项的和表示用最优冲激响应 $h_0(t)$ 时产生的均方误差，用 MSE_0 表示这三项的和。由于自相关函数 $R_x(u-t)$ 是偶函

数，所以式（7.4.16）的第四项和第五项是相等的，把它们与第六项组合起来，将方程的形式改写为如下的形式：

$$\begin{aligned}MSE &= MSE_0 + 2\int_{\infty} g(u)\left[\int_{\infty} h_0(\tau)R_x(u-\tau)d\tau - R_{xs}(u)\right]du \\ &\quad + \int_{\infty}\int_{\infty} g(u)g(\tau)R_x(u-\tau)dud\tau \\ &= MSE_0 + T_4 + T_5\end{aligned} \quad (7.4.17)$$

这里引入 T_4 和 T_5 仅仅是为了表达简洁。

现在证明 T_5 是非负的。将自相关函数 $R_x(u-t)$ 写为积分形式：

$$T_5 = \int_{\infty}^{\infty}\int_{\infty}^{\infty} g(u)g(\tau)\int_{\infty} x(t-\tau)x(t-u)dtdud\tau \quad (7.4.18)$$

变换积分次序得到：

$$T_5 = \int_{\infty}^{\infty}\int_{\infty}^{\infty} g(u)x(t-u)du\int_{\infty}^{\infty} g(\tau)x(t-\tau)d\tau dt \quad (7.4.19)$$

定义 $z(t)$ 为 $g(t)$ 和 $x(t)$ 的卷积函数，那么式（7.4.18）就可以写为：

$$T_5 = \int_{\infty}^{\infty} z^2(t)dt \geqslant 0 \quad (7.4.20)$$

其中，$z(t) = \int_{\infty} g(u)x(t-u)du$。

于是，（7.4.17）式可写为：

$$MSE = MSE_0 + 2\int_{\infty} g(u)\left[\int_{\infty} h_0(\tau)R_x(u-\tau)d\tau - R_{xs}(u)\right]du + T_5 \quad (7.4.21)$$

其中，MSE_0 是最优条件下的均方误差，T_5 独立于 h_0 且非负。

要找出使 MSE_0 是 MSE 能达到最小值时 $h_0(\tau)$ 应满足的条件，一种方法是让方括号中的式子对所有的 u 值都为 0。这样可以去掉式（7.4.17）中的 T_4 项，而且保证了 $MSE_0 \leqslant MSE$。尽管施加这个条件看上去正是要做的事情，但是还是必须保证它是使滤波器最优的充要条件。可以推导必要性：假定对于某个 u 值，式（7.4.21）中方括号的项为非零值。那么，因为 $g(u)$ 是任意函数，当括号里的项是正值时，$g(u)$ 可以是绝对值更大的负值，反过来也是这样。T_4 中的积分将是很大的负值，这样 MSE 就会比 MSE_0 更小。这样和定义矛盾，因此可以得出结论，式（7.4.21）中括号里的项等于 0 是必要的。这就是说这里得到了一个最优滤波器的必要条件：

$$R_{xs}(\tau) = \int_{\infty} h_0(u)R_x(u-\tau)du \quad (7.4.22)$$

因此，式（7.4.14）带来的复杂化由于得到了最优滤波器的必要条件而得到补偿。

同时，式（7.4.22）也是使滤波器最优化的充分条件，这里不再需要其他的条件。既然必要条件可以将式（7.4.17）简化，于是式（7.4.17）变为：

$$MSE = MSE_0 + T_5 \quad , \quad T_5 \geqslant 0 \quad (7.4.23)$$

显然：

$$MSE \geqslant MSE_0 \quad (7.4.24)$$

因此，（7.4.22）式从均方意义上定义了最优线性估值器的冲激响应。

由（7.4.22）式得：

$$R_{xs}(\tau) = h_0(u) * R_x(u) \quad (7.4.25)$$

这就把最优冲激响应跟输入信号的自相关以及输入信号和期望信号的互相关等关联

起来了。式（7.4.25）中的第二个等号来自式（7.4.24），它说明了维纳滤波器使得输入/输出的互相关函数等于信号/信号加噪声的互相关函数。

对（7.4.25）式做傅立叶变换，得到：

$$P_{xs}(s) = H_0(s)P_x(s) \quad (7.4.26)$$

故：

$$H_0(s) = \frac{P_{xs}(s)}{P_x(s)} \quad (7.4.27)$$

是维纳滤波器的频域特征。

4. 维纳滤波器设计

由（7.4.27）式，只要知道一个输入信号 $s(t)$，录得 $x(t)$，知道 $P_{xs}(s)$ 和 $P_x(s)$ 的函数形式（白噪声），则可以计算 $H_0(s)$，从而可计算最优线性估值器的冲激响应 $h_0(t)$：

$$h_0(t) = \int_{\infty}^{\infty} H_0(s)e^{j2\pi st}ds \quad (7.4.28)$$

具体过程如下：

（1）对输入信号 $s(t)$ 的样本进行数字化。

（2）求出输入样本的自相关得到 $R_x(\tau)$ 的一个估计值。

（3）计算 $R_x(\tau)$ 的傅立叶变换得到 $P_x(s)$。

（4）在无噪声的情况下对取得输入信号的一个样本并进行数字化。

（5）求信号样本（即无噪声情况下的）与输入样本的互相关来估计 $R_{xs}(\tau)$。

（6）计算 $R_{xs}(\tau)$ 的傅立叶变换得出 $P_{xs}(s)$。

（7）从式（7.4.27）计算最优滤波器的传递函数。

（8）如果要用卷积运算来实现滤波器，则计算 $H_0(s)$ 的傅立叶逆变换得到最优线性估值器的冲激响应 $h_0(t)$。

如果取得无噪声信号和输入信号的样本不可能或者不现实，那么可以为式（7.4.27）中的自相关函数或功率谱设定一种函数形式。例如，"白噪声"有恒定的功率谱，也可以设定其他的函数形式或它的功率谱。

7.4.3 维纳滤波器举例

1. 如果信号与噪声不相关

式（7.4.25）中的自相关函数和式（7.4.27）中的功率谱不够直观，也较难理解。然而，如果假定噪声与信号是不相关的，那情形就好多了。

从定义上说，即：

$$\varepsilon\{s(t)n(t)\} = \varepsilon\{s(t)\}\varepsilon\{n(t)\} \quad (7.4.29)$$

对（7.4.27）式的分子变换为：

$$R_{xs}(\tau) = \varepsilon\{(x(t)s(t+\tau)\} = \varepsilon\{[s(t)+n(t)]s(t+\tau)\} \quad (7.4.30)$$

或者：

$$R_{xs}(\tau) = \varepsilon\{s(t)s(t+\tau)\} + \varepsilon\{n(t)s(t+\tau)\} \quad (7.4.31)$$

根据式（7.4.29），进一步得到：

$$R_{xs}(\tau) = R_s(\tau) + \varepsilon\{n(t)\}\varepsilon\{s(t+\tau)\} = R_s(\tau) + \int_{\infty}^{\infty} n(t)dt\int_{\infty}^{\infty} s(t+\tau)dt \quad (7.4.32)$$

于是：

$$R_{xs}(\tau)=R_s(\tau)+N(0)S(0) \tag{7.4.33}$$

其中：

$$N(0)=\int_{-\infty}^{\infty}n(t)e^{-j2\pi 0t}dt,\qquad S(0)=\int_{-\infty}^{\infty}s(t)e^{-j2\pi 0t}dt \tag{7.4.34}$$

对（7.4.27）分母作类似变换：

$$R_x(\tau)=R_s(\tau)+R_n(\tau)+2S(0)N(0) \tag{7.4.35}$$

于是（7.4.27）变为：

$$H_0(s)=\frac{P_s(s)+N(0)S(0)\delta(s)}{P_s(s)+P_n(s)+2N(0)S(0)\delta(s)} \tag{7.4.36}$$

或者，忽略0频率项：

$$H_0(s)=\begin{cases}\dfrac{P_s(s)}{P_s(s)+P_n(s)}, & s\neq 0\\[2mm] \dfrac{1}{2}, & s=0\end{cases} \tag{7.4.37}$$

因为，$\delta(s)=0$。当 $s\neq 0$ 时，且 $\delta(0)$ 无限大，所以 $N(0)\neq 0$，$S(0)\neq 0$。

由（7.4.37）式：

当 $P_n(s)=0$ 时，$H_0(s)=1$，全部信号通过。

当 $P_s(s)=0$ 时，$H_0(s)=0$，抑制噪声。

当 $P_s(s)=P_n(s)$ 时，$H_0(s)=\dfrac{1}{2}$。

2. 滤波器性能（误差估计）

回忆式（7.4.22）给出的最优必要条件代入式（7.4.13）的第三项，它与第二项组合可以得到：

$$MSE_0=R_s(0)-\int_{-\infty}^{\infty}h_0(\tau)R_{xs}(\tau)d\tau \tag{7.4.38}$$

若 $n(t)$ 与 $s(t)$ 无关，且 $\varepsilon\{n(t)\}=0$，则由（7.4.32）式代入（7.4.38）式得：

$$MSE_0=R_s(0)-\int_{-\infty}^{\infty}h_0(\tau)F^{-1}\{P_s(s)\}d\tau \tag{7.4.39}$$

展开逆变换并改变积分次序得到：

$$MSE_0=R_s(0)-\int_{-\infty}^{\infty}P_s(s)\int_{-\infty}^{\infty}h_0(\tau)e^{j2\pi s\tau}d\tau ds \tag{7.4.40}$$

把第一项和第二个积分看作是傅立叶变换，于是：

$$MSE_0=\int_{-\infty}^{\infty}P_s(s)ds-\int_{-\infty}^{\infty}P_s(s)H_0(-s)ds \tag{7.4.41}$$

当若 $n(t)$ 与 $s(t)$ 无关，且 $\varepsilon\{n(t)\}=0$ 时，$H_0(s)$ 是偶函数，$H_0(-s)=H_0(s)$，故：

$$MSE_0=\int_{-\infty}^{\infty}P_s(s)ds-\int_{-\infty}^{\infty}P_s(s)\frac{P_s(s)}{P_s(s)+P_n(s)}ds \tag{7.4.42}$$

或写为：

$$MSE_0=\int_{-\infty}^{\infty}\frac{P_s(s)P_n(s)}{P_s(s)+P_n(s)}ds=\int_{-\infty}^{\infty}P_n(s)H_0(s)ds \tag{7.4.43}$$

3. 维纳滤波器的传递函数

图 7-18 用图示方式说明了在信号与噪声无关情况下维纳滤波器的频域行为。在低频时，信号能量远远大于噪声能量，传递函数的值趋近于 1，使信号能量通过；在信号和噪声功率相等处传递函数的值降到 0.5；在更高频的区域（这时噪声占主导地位），传递函数的值逐渐到零。

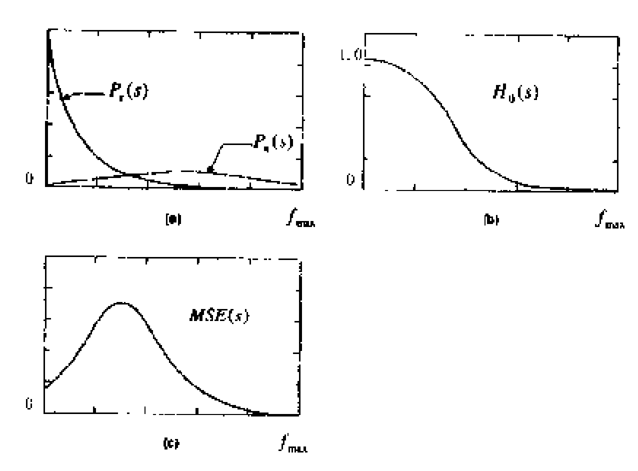


图 7-18 维纳滤波器的传递函数

当假定自己知道 $s(t)$ 和 $n(t)$ 的功率谱时，应承认自己对信号的噪声的相位一无所知。

注意： 传递函数 $H_0(s)$ 是实偶函数，因此没有引入相移。

表明滤波器从噪声中恢复信号的本领的指标——输出端的实际均方误差由式（7.4.43）给出，图 7-18 中画出了其积分值。从图上可以看出对 MSE 有贡献的频段是有限的，位于信号和噪声的功率谱都非零的频段。图 7-19 表示信号和噪声在频域里可分开的情况。在这种情况下，维纳滤波能让信号全都通过而彻底地抑制噪声。

图 7-20 表示信号带宽有限，且包含在白噪声中的情况。$H_0(s)$ 是带通滤波器，如果信号功率谱恒定的，那么均方误差就与带宽成比例。

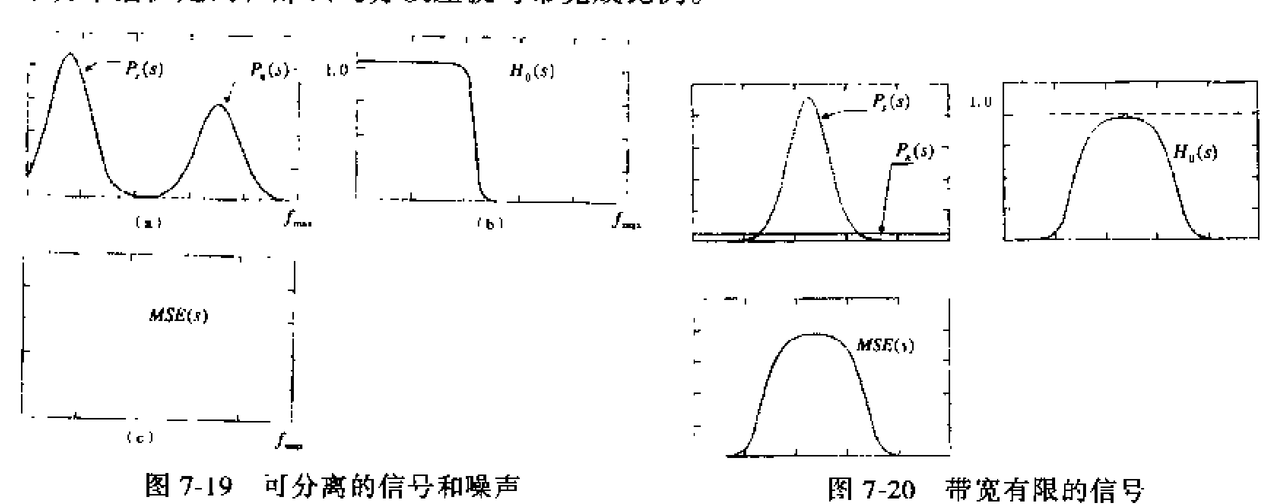


图 7-19 可分离的信号和噪声

图 7-20 带宽有限的信号

如果信噪比很低，式（7.4.43）可以近似为：

$$MSE_0 \approx \int_{-\infty}^{\infty} P_s(s)ds = \int_{-\infty}^{\infty} |S(s)|^2 ds \tag{7.4.44}$$

根据 Rayleigh 定理：

$$MSE_0 \approx \int_{-\infty}^{\infty} S^2(t)ds = R_s(0) = energy \tag{7.4.45}$$

因此在这种情况下，均方误差与信号的能量成正比。

如前所述，一般的去卷积并不考虑噪声，因此当有噪声时，去卷积传递汉化素（在高频时它的幅度值常常很大）是不实用的。图 7-21 是一个二维维纳去卷积的例子。这里模糊函数是高斯型的，信号功率谱是 $1/f$，而且是白噪声的情况。

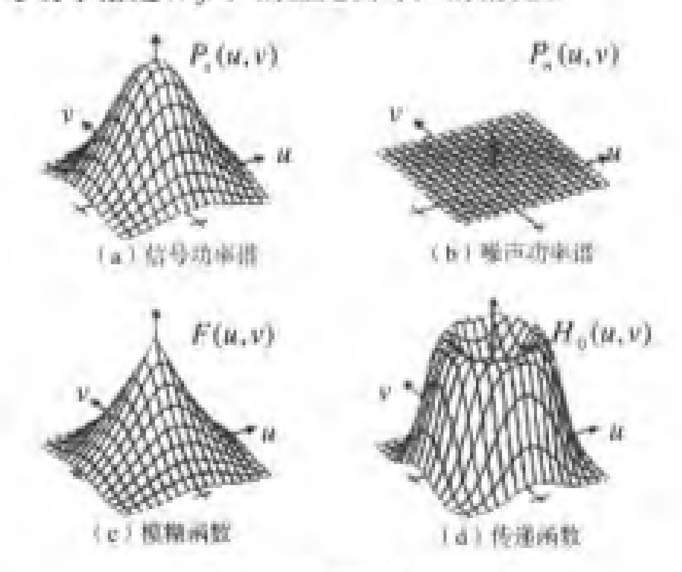
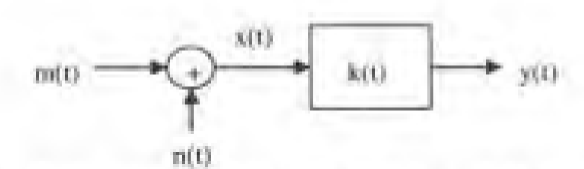


图 7-21 二维维纳去卷积举例

7.4.4 匹配检测器（Matched Detector）

维纳滤波器主要用于从噪声污染的信号中恢复未知的原信号。匹配检测器被用来从复杂的背景中找出已知信号的位置。匹配检测器的条件是知道原信号的形状，但不知什么时候出现。

模型：



目的是求 $k(t)$，使得通过 $y(t)$ 能方便地检测出已知形式的信号。

例如，如图 7-22 所示为雷达系统。

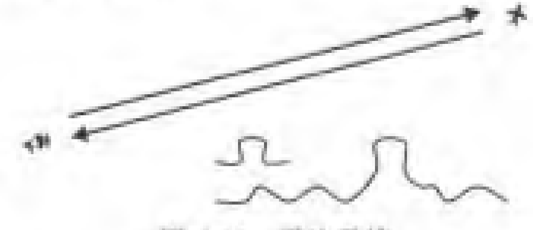


图 7-22 雷达系统

假定 $m(t)$ 和 $n(t)$ 是遍历性随机变量，且功率谱是已知的。

$$y(t)=[m(t)+n(t)]*k(t)=m(t)*k(t)+n(t)*k(t)=u(t)+v(t) \tag{7.4.46}$$

定义两个分量的输出为：

$$u(t)=m(t)*k(t) \qquad v(t)=n(t)*k(t) \tag{7.4.47}$$

现在 $u(t)$ 就是滤波之后的信号，而 $v(t)$ 则是滤波之后的噪声。

跟讨论维纳滤波器一样，首先必须规定要有多少关于信号和噪声的知识，并建立最优化准则。假定知道 $m(t)$ 的函数形式，但不知道信号将会在哪个时间点出现。匹配检测器的经典应用就是检测反射回来的雷达脉冲，在这种情况下，反射回来的脉冲与反射出去的脉冲在形式上是一样的，但是不知道什么时候出现。在数字图像处理中，匹配检测器对于定位有噪声的图像中已知特征是很有用的。

跟讨论维纳滤波器一样，假定噪声是遍历性的随机变量且功率谱是已知的。下面希望设计出能通过观察输出而达到最优地检测出信号 $k(t)$。

最优化准则：把输出端的平均信号功率比在 0 时刻的取值作为衡量滤波器的一种度量。

$$\rho=\frac{\varepsilon\left\{u^2(t)\right\}}{\varepsilon\left\{v^2(t)\right\}} \tag{7.4.48}$$

其中，ρ 是输出端的平均信噪比在 t 时刻的取值。信号功率越大，越容易检测。

原型信号 $m(t)$ 常常是以原点为中心的相当窄的函数。若 $m(t-t_1)$，m 在 t_1 出现，则在 t_1 时刻 $y(t)$ 的幅度增大。

显然，如果 ρ 很大，那么输出 $y(t)$ 的幅度将很大程度上依赖于 $m(t)$ 是否出现，并且对噪声 $n(t)$ 的波动不敏感。因此选择使 ρ 最大作为最优化准则。

因为 u(t)是固定不变的，所以 $\varepsilon\left\{u^2(t)\right\}=u^2(0)$，因此可以去掉分子中的期望算子，并把式（7.4.48）改写为：

$$\rho=\frac{u^2(0)}{\varepsilon\left\{v^2(t)\right\}}=\frac{\left[m(t)*k(t)\right]^2\Big|_{t=0}}{\varepsilon\left\{\left[n(t)*k(t)\right]^2\right\}}=\frac{\left[F^{-1}\left\{M(s)K(s)\right\}\right]^2\Big|_{t=0}}{\varepsilon\left\{\left[n(t)*k(t)\right]^2\right\}}=\frac{\rho_n}{\rho_d} \tag{7.4.49}$$

这里引入 ρ_n 和 ρ_d 后，就可以分别地考虑分子和分母。

可以证明：

$$\rho=\frac{\left[\int_{-\infty}^{\infty}K(s)M(s)ds\right]^2}{\int_{-\infty}^{\infty}\left|K(s)\right|^2P_n(s)ds} \tag{7.4.50}$$

考虑分子，由于：

$$F\left\{m(t)*k(t)\right\}=M(s)K(s) \tag{7.4.51}$$

因此：

$$\begin{aligned}\left[m(t)*k(t)\right]^2\Big|_{t=0}&=\left[F^{-1}\left\{M(s)K(s)\right\}\right]^2\Big|_{t=0}\\&=\left[\int_{-\infty}^{\infty}(M(s)K(s)e^{+j2\pi st})\Big|_{t=0}ds\right]^2\\&=\left[\int_{-\infty}^{\infty}M(s)K(s)ds\right]^2\end{aligned} \tag{7.4.52}$$

分母：

$$\rho_d=\varepsilon\{\int_{-\infty}^{\infty}k(q)n(t-q)dq\int_{-\infty}^{\infty}k(\tau)n(t-\tau)d\tau\} \tag{7.4.53}$$

分母可以写为：

$$\rho_d=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}k(q)k(\tau)\varepsilon\{n(t-q)n(t-\tau)\}dqd\tau \tag{7.4.54}$$

其中：

$$\varepsilon\{n(t-q)n(t-\tau)\}=R_n(\tau-q)=\int_{-\infty}^{\infty}P_n(s)e^{j2\pi s(\tau-q)}ds$$

于是：

$$\begin{aligned}\rho_d&=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}k(q)k(\tau)\int_{-\infty}^{\infty}P_n(s)e^{j2\pi s(\tau-q)}ds\,dq\,d\tau\\&=\int_{-\infty}^{\infty}P_n(s)[\int_{-\infty}^{\infty}k(q)e^{-j2\pi qs}dq\int_{-\infty}^{\infty}k(\tau)e^{j2\pi\tau s}d\tau]ds\\&=\int_{-\infty}^{\infty}P_n(s)[K(s)K(-s)]ds\\&=\int_{-\infty}^{\infty}\left|K(s)\right|^2P_n(s)ds\end{aligned}\qquad(7.4.55)$$

故式（7.4.50）$\rho=\dfrac{\left[\int_{-\infty}^{\infty}K(s)M(s)ds\right]^2}{\int_{-\infty}^{\infty}\left|K(s)\right|^2P_n(s)ds}$ 得证。

以下证明：$\rho\leqslant\int_{-\infty}^{\infty}\dfrac{\left|M(s)\right|^2}{P_n(s)}ds$

并且当 $K_0(s)=C\dfrac{M^*(s)}{P_n(s)}$ 时，

$\rho_{\max}=\int_{-\infty}^{\infty}\dfrac{\left|M(s)\right|^2}{P_n(s)}ds$ 。

由施瓦兹不等式：

$$\left[\int f(s)g(s)ds\right]^2\leqslant\int f^2(s)ds\int g^2(s)ds\qquad(7.4.56)$$

等号成立当且仅当 $f(s)=cg^*(s)$， c 为常数，*为共轭。

令：

$$f(s)=K(s)\sqrt{P_n(s)}\qquad(7.4.57)$$

$$g(s)=\frac{M(s)}{\sqrt{P_n(s)}}\qquad(7.4.58)$$

代入（7.4.56）中，可以证明：

$$\left[\int K(s)M(s)ds\right]^2\leqslant\int(K(s)\sqrt{P_n(s)})^2ds\cdot\int(\frac{M(s)}{\sqrt{P_n(s)}})^2ds\qquad(7.4.59)$$

$$\rho=\frac{\left[\int_{-\infty}^{\infty}K(s)M(s)ds\right]^2}{\int_{-\infty}^{\infty}\left|K(s)\right|^2P_n(s)ds}\leqslant\int_{-\infty}^{\infty}\frac{\left|M(s)\right|^2}{P_n(s)}ds\qquad(7.4.60)$$

ρ 有界，在 $f(s)=cg^*(s)$ 时达到上界。

即当：

$$K_0(s)=C\frac{M^*(s)}{P_n(s)}\qquad(7.4.61)$$

时，ρ 达到最大。

$$\rho_{\max}=\int_{-\infty}^{\infty}\frac{\left|M(s)\right|^2}{P_n(s)}ds\qquad(7.4.62)$$

$K_0(s)$ 是最佳检测滤波器的传递函数。

7.4.5 匹配检测器举例

1. 白噪声

第一种情况：假定噪声的频谱是白的，即：

$$P_n(s) = N_0^2 \tag{7.4.63}$$

令 $C = N_0^2$，由（7.4.61）式，匹配检测器变为：

$$K_0(s) = M^*(s) \tag{7.4.64}$$

在时域里，冲激响应为：

$$k_0(t) = F^{-1}\{K_0(S)\} = \int_{-\infty}^{\infty} M^*(s)e^{j2\pi st}ds \tag{7.4.65}$$

因为 $m(t)$ 为实型，所以 $M(s)$ 为 Hermite 函数，即：

$$k_0(t) = \int_{-\infty}^{\infty} M(-s)e^{j2\pi(-s)(-t)}ds = m(-t) \tag{7.4.66}$$

因此对于白噪声的情况，冲激响应就是信号本身的反转。此滤波器被认为已和信号匹配了，或者说是一个匹配的滤波器。

输出端的信号分量为：

$$u(t) = m(t) * k_0(t) = \int_{-\infty}^{+\infty} m(\tau)m(-t+\tau)d\tau = R_m(-t) \tag{7.4.67}$$

而噪声分量为：

$$v(t) = n(t) * k_0(t) = \int_{-\infty}^{+\infty} n(\tau)n(-t+\tau)d\tau = R_{mn}(-t) \tag{7.4.68}$$

因为式（7.4.67）和式（7.4.68）中 $k_0(t)$ 正好是要检测的信号的反转，所以匹配滤波器 $k_0(t)$ 只是一个互相关算子，它把进来的信号加上噪声与已知形式的期望信号作相关运算，所得输出为：

$$y(t) = u(t) + v(t) = m(t) * k_0(t) + n(t) * k_0(t) = R_m(-t) + R_{mn}(-t) \tag{7.4.69}$$

此输出在任何地方都有一个互相关分量，但是只有在信号出现的地方才有自相关分量。如果噪声与信号之间的相关性很小，那么对所有的 τ 值，$R_{mn}(\tau)$ 都很小，输出的噪声分量也很小。

不仅如此，自相关函数 $R_m(\tau)$ 在 $\tau = 0$ 时出现峰值，所以：

$$\rho = \frac{u^2(0)}{\varepsilon\{v^2(t)\}} \tag{7.4.70}$$

在 $t = 0$ 处，（或其他信号出现的地方）变得很大，这正是人们所期望的。

2. 矩形脉冲检测器

假定 $m(t) = \Pi(t)$，即要求设计用来检测淹没在白噪声中的矩形脉冲的匹配检测器。假定输入信号是 $x(t) = s(t) + n(t)$，其中 $s(t) = \Pi(t-T)$，$n(t)$ 为白噪声。矩形脉冲的自相关函数为：

$$R_x(\tau) = \Pi(t) * \Pi(t) = \Lambda(\tau)$$

则：

$$u(t) = m(t) * k_0(t) = \Pi(t) * \Pi(t-T) = \Lambda(t-T)$$

又 $v(t) = R_{mn}(t)$

则：

$y(t)=u(t)+v(t)=\Lambda(t-T)+R_{mn}(t)$

注意：匹配检测器并不能保持信号的形式或形状，这一点与维纳滤波器是不同的。这是因为设计的滤波器是用来检测待定的已知输入信号是否出现，而不是检测无噪声时信号的形式或形状。

7.4.6 通过求平均值降噪

如何衡量一个图像的好坏呢？这里用到了功率信噪比。

对于一幅图像，定义二维图像为：

$D(x,y)=S(x,y)+N(x,y)$

设 $S(x,y)$ 为理想图像，$N(x,y)$ 为噪声，则功率信噪比为：

$$P(x,y)=\frac{S^2(x,y)}{N^2(x,y)} \tag{7.4.71}$$

功率性噪比 $P(x,y)$ 越大，则图像越好。由上式可以很直观的看出，理想图像的平方值与噪声的平方比值越大，证明噪声能量相对于理想图像越小，则对图像的质量影响越小。

一个静止场景的 M 幅图像，假设 M 幅像为：

$$D_i(x,y)=S(x,y)+N_i(x,y) \qquad i=1,2,\dots,M \tag{7.4.72}$$

假定每幅噪声图像来自于同一个互不相关的，噪声均值等于 0 的随机噪声图像的样本集，则单幅图像的功率信噪比为：

$$P(x,y)=\frac{S^2(x,y)}{\varepsilon\{N^2(x,y)\}} \tag{7.4.73}$$

并且：

$$\varepsilon\{N_i(x,y)\}=0 \quad \varepsilon\{N_i(x,y)+N_j(x,y)\}=\varepsilon\{N_i(x,y)\}+\varepsilon\{N_j(x,y)\} \ (i\neq j) \tag{7.4.74}$$

$$\varepsilon\{N_i(x,y)N_j(x,y)\}=\varepsilon\{N_i(x,y)\}\varepsilon\{N_j(x,y)\} \qquad (i\neq j) \tag{7.4.75}$$

$\varepsilon\{\ \}$ 为期望，$\varepsilon\{N_i(x,y)\}$ 为在样本集中所有噪声图像在点 (x,y) 处的平均值。

由（7.4.71）、（7.4.72）得平均值：

$$\overline{D}(x,y)=\frac{1}{M}\sum_{i=1}^{M}\left[S(x,y)+N_i(x,y)\right]$$

$$\overline{P}(x,y)=\frac{S^2(x,y)}{\varepsilon\left\{\left[(\frac{1}{M})\sum_{i=1}^{M}N_i(x,y)\right]^2\right\}}$$

化简：

$$\overline{P}(x,y)=\frac{M^2S^2(x,y)}{\varepsilon\left\{\sum_{i=1}^{M}\sum_{j=1}^{M}N_i(x,y)N_j(x,y)\right\}}$$

利用互不相关性，展开分母，得：

$$\overline{P}(x,y)=\frac{M^2S^2(x,y)}{\varepsilon\left\{\sum_{i=1}^{M}N_i^2(x,y)\right\}+\varepsilon\left\{\sum_{i=1}^{M}\sum_{j=1}^{M}N_i(x,y)N_j(x,y)\right\}}$$

$$= \frac{M^2 S^2(x,y)}{\sum_{i=1}^{M} \varepsilon\left\{N_i^2(x,y)\right\} + \sum_{i=1}^{M}\sum_{j=1}^{M} \varepsilon\{N_i(x,y)\}\varepsilon\left\{N_j(x,y)\right\}}$$

故：

$$\overline{P}(x,y) = \frac{M^2 S^2(x,y)}{M\varepsilon\left\{N^2(x,y)\right\}} = MP(x,y) \tag{7.4.76}$$

即，对 M 幅图像进行平均，使图像中每一点的平方信噪比提高了 M 倍。当然 M 越大越好。幅度信噪比为：

$$\overline{SNR} = \sqrt{\overline{P(x,y)}} = \sqrt{M}\sqrt{P(x,y)} \tag{7.4.77}$$

7.4.7 一个实例（血管造影术）

下面通过举例说明如何应用最优滤波器理论来进行实际的滤波器设计。如图 7-23 所示，对于一个血管的 X 射线图，要测量血管直径的大小。该图像用来模型化血管造影技术。血管造影技术是在 X 射线照射期间把某种染色颜料注射入血管的一种诊断技术。在这里，用光滑的导管来代替血管进行实验。

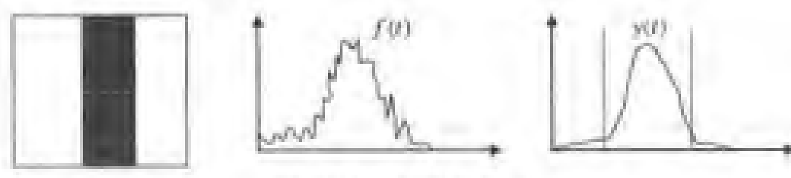


图 7-23　血管造影术

本例的目标是开发一种处理技术，利用该技术将找到有噪声图像中导管的边界，并可靠地测出导管的直径。该技术可用于定量地分析伴有粥样硬化和导致心脏病发作地血管狭窄症。由于这是一个边缘检测问题，匹配检测器可以采用，但本例的提法不同。这里试图通过维纳滤波器估计出原信号，然后通过微分的方法求边缘。假定在每一条图像扫描线上，血管的边界出现在斜率最大的点上，并尝试用微分的办法给边界定位。这里将单独地处理每一条水平扫描线，这样把问题简化为一维，而且允许处理过程对宽度上的急剧变换做出响应。

如上图 7-23 所示，图像中一条扫描线 $f_i(t)$ 的灰度级曲线。主要由于底片颗粒和光子统计特性的缘故，图中的噪声是很明显的。显然，对这条曲线做微分运算是不会在拐点处产生可靠的尖峰，这是因为图像有严重的噪声。对于一条水平扫描线，血管的边界为斜率最大的点，用微分的方法可以求边缘，但必须先去噪。其解决方案如下：

（1）首先去噪。用平均去噪法估计信号功率谱 $P_s(s)$ 和噪声功率谱 $P_n(s)$。假设 $s(x)$ 和噪声 $n(x)$ 是不相关的，维纳滤波器要求知道信号的功率谱以及噪声的功率谱。可以通过对多条扫描线求平均来估计信号的功率谱，这样利用平均降噪得到的信号可以近似于原信号。这是由于导管是平滑的，所有扫描线 $f_i(x)$ 在没有噪声的情况下应该是相同的，所以：

$$s(x) \approx \frac{1}{N}\sum_{i=1}^{N} f_i(x)$$

$$P_s(s) = \left|F\left\{s(x)\right\}\right|^2 \approx \left|F\left\{\frac{1}{N}\sum_{i=1}^{N} f_i(x)\right\}\right|^2$$

$n_i(x) \approx f_i(x) - s(x)$

$P_n(s) \approx \frac{1}{N}\sum_{i=1}^{N}\left|F\{f_i(x)-s(x)\}\right|^2$

（2）利用 Wiener Filter 滤波，其传递函数 $H_0(s)$ 如下：

$$H_0(s) = \frac{P_s(s)}{P_s(s)+P_n(s)} \tag{7.4.78}$$

通过傅立叶反变换得到冲激响应 $h_0(x)$，然后卷积得到滤波后的输出：

$h_0(x) = \mathcal{F}^{-1}\{H_0(s)\}$

$s(x) \approx y(x) = f_i(x) * h_0(x)$

（3）用微分方法求边缘：求两个导数的极值点。

（4）求直径：两个导数的极值点之间的距离就是直径。

这样就可以求出血管的直径了。

注意：这里得到的 $H_0(s)$，如图 7-24（a）所示，图中的曲线有凹口是由信号功率谱的零交叉点产生的。根据相似性定理，这些凹口的位置伴随着导管宽度的变化而发生平移。

这一点说明一个事实：信号并非一个遍历性的随机过程。在信号集合中不同的成员函数对应于不同宽度的导管，因而并不是都有相同的功率谱。既然如此，这里就不得不违反维纳滤波器的一条基本假设。依旧这样做下去，因为在最优无法达到的情况下，“次最优”技术可以作为真实最优的替代。

$\tilde{H}(s)$ 为 $H_0(s)$ 的平滑近似，如图 7-24（b）所示。

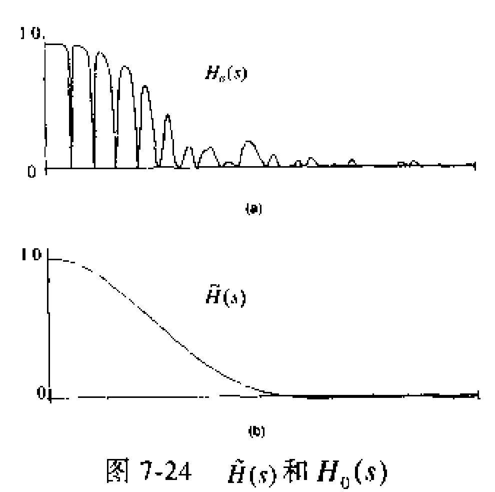


图 7-24　$\tilde{H}(s)$ 和 $H_0(s)$

（a）维纳滤波器的传递函数。

（b）是（a）的平滑近似。

又因为：

$\tilde{h}(x) = \mathcal{F}^{-1}\{\tilde{H}(s)\}$

因此：

$y(x) = f_i(x) * \tilde{h}(x)$

于是：

$\frac{dy(x)}{dx} \approx \frac{d}{dx}[f_i(x) * \tilde{h}(x)] = f_i(x) * \tilde{h}'(x)$

找使能$|y'(x)|$极大的两个坐标点 x_1 和 x_2，即为血管的边缘。如图 7-25 所示。

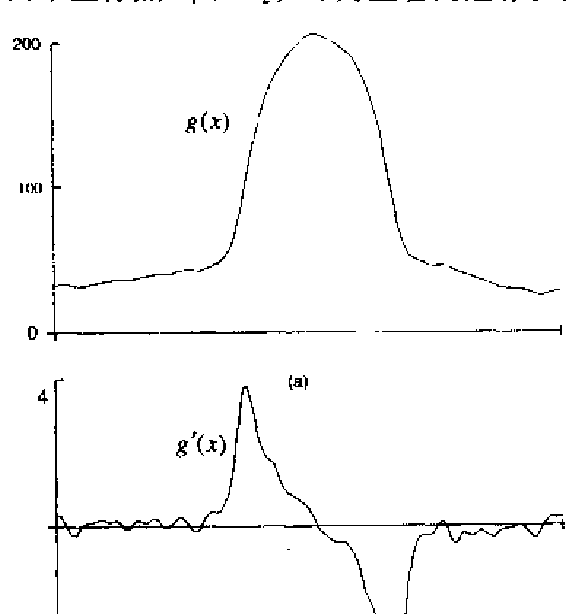


图 7-25 血管的边缘检测

7.5 排序统计滤波器

排序统计滤波器是一种非线性的、非卷积滤波器。人们已经提出、测试并使用了很多种非线性滤波器。可以说明，非线性滤波器能够比线性滤波器更好地解决某些数字图像处理问题。然而这些非线性滤波器缺乏像线性滤波器那样透彻的且相对直接的理论背景。这里只讨论其中的一种最有用的——排序统计滤波器。排序统计滤波器基于对集合中的元素排序结果进行滤波，是基于对集合中的元素排序以后得到的统计量而不是计算均值、方差等。中值滤波器就是其中的一种。

7.5.1 中值滤波器

中值滤波器（如图 7-26 所示）是最常用的非线性滤波器。中值滤波器类似低通线性滤波器用于去噪，但中值滤波器在衰减噪声的同时不使边界模糊。它是一种邻域计算，类似于卷积，但是计算不是加权求和，而是把邻域中的像素按灰度级进行排序，然后选择该组的中间值作为输出像素值。

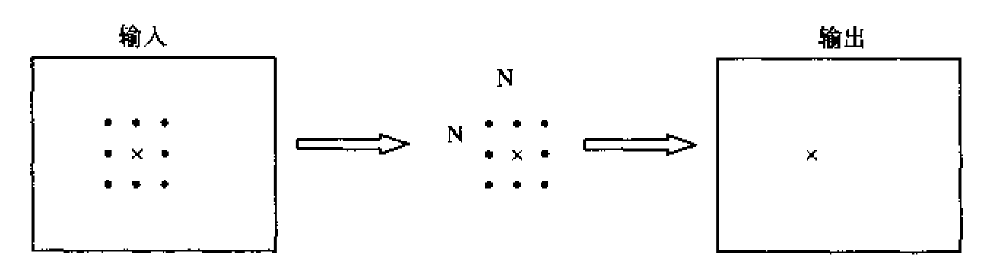


图 7-26 中值滤波器

基本原理：将数字图像上每个点的值用该点的一个邻域中各点值的中值代替。

中值的定义如下：

假设邻域内有几个点 $x_{i1}, x_{i2}, \ldots, x_{in}$，把 n 个数按值的大小顺序排列如下：

$x_{i1} \leqslant x_{i2} \leqslant x_{i3} \leqslant \cdots \leqslant x_{in}$

$$y_i = Med\{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{in}\} = \begin{cases} x_{i(\frac{n+1}{2})} & n\text{为奇数} \\ \frac{1}{2}[x_{i(\frac{n}{2})} + x_{i(\frac{n}{2}+1)}] & n\text{为偶数} \end{cases}$$

则 y_i 称为序列 $x_{i1}, x_{i2}, \ldots, x_{in}$ 的中值。

例如，序列 (80,90,200,110,120) 的中值是 110。

7.5.2 滤波模板因子

中值滤波器去噪声的效果依赖于两个相关的但完全分开的要素：邻域的空间范围及中值计算涉及的像素数。较简单的情况是一个 $N \times N$ 的方形模版（N 常常是奇数），计算时用到所有 N^2 个像素点。

这里也可以使用如图 7-27 所示的稀疏分布的模版来节省时间。

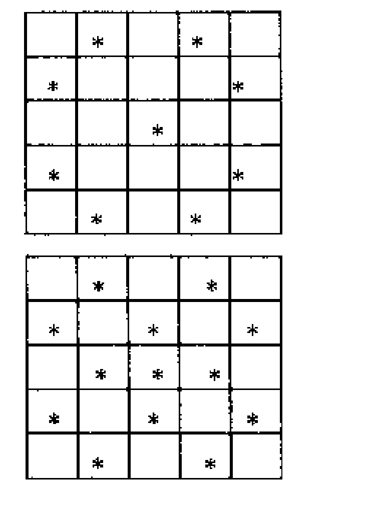
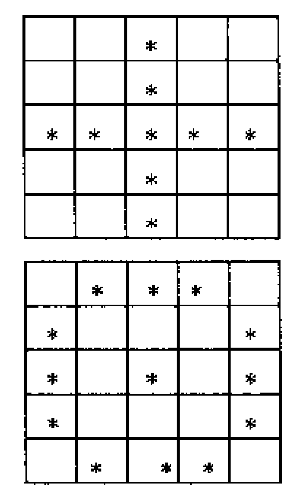
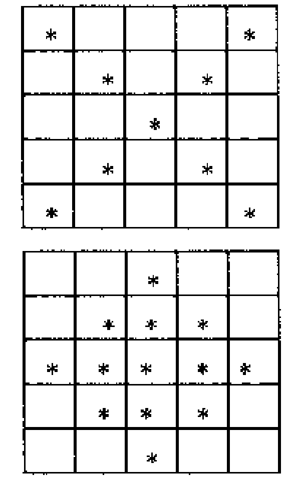




图 7-27 稀疏分布的模版

7.5.3 其他排序统计滤波器——第 X 百分点的滤波器

中值代表序列中第 50 个百分点的数值；第 1 个和第 100 个百分点可用于最小值和最大值滤波器；同理，可以构造第 X 百分点的滤波器，通常，$X < 50$ 时，滤波器趋于使图像变暗；$X > 50$ 时，滤波器趋于使图像变亮。

尽管这些滤波器对一般的应用不太普及，但是在某些特殊的应用中，它们是有价值的。

小结

本章主要介绍了低通滤波器、带通或带阻滤波器、高频增强滤波器、最优线性滤波器设计、排序统计滤波器等内容。

重点：本章重点主要有以下几个方面：

（1）高频增强滤波器的传递函数有一个零频率效率响应，它等于两个分量脉冲下的面积的差。

（2）滤波器的零频率响应决定了大特征的对比度受影响的程度。

（3）按计算简便而不是性能最优而设计的滤波器可能给图像引入一些人工痕迹。

（4）遍历性随机过程是指这样的一种信号，该信号的已知的功率谱和自相关函数代表了关于该信号的全部知识。

（5）从均方误差的意义上讲，对于从功率谱已知的加性噪声中恢复功率谱已知的信号这样一类任务来说，维纳滤波器是最优的。

（6）维纳滤波器的传递函数在高信噪比的频段取值接近于1。而在噪声起支配作用的频段，其取值接近于0。

（7）对于从加性噪声背景中检测已知的信号是否出现这样一类任务来说，匹配检测器是最优。

（8）维纳滤波器传递函数是实偶且有界（取值位于0和1之间）。

（9）一般来说，匹配滤波器传递函数是复数、Hermite型的，且是无界。

（10）排序统计滤波器是非线性的，并通过给邻域中的像素排序来进行滤波。

（11）中值滤波器可以消除比它的尺度小一半的物体，而保留较大的物体。它对于既保留边缘又要去噪声的处理很有用。

（12）稀疏分布的模板能减少大尺度的中值滤波器所需的计算时间。

难点： 掌握维纳滤波器和匹配检测器的原理，以及最优滤波器设计。

疑点： 当人们在一个域里使用方形滤波器时，常常受到在另一个域里出现的减幅振荡（振铃现象）和其他的人工效应所困扰。如何在最优性和计算复杂性之间进行权衡和取舍而不会招致灾难性的人工效应。

练习七

一、选择题

1. 如果要设计一个滤波器是的信号通过时保留细节部分，请问用（ ）滤波器。
   A. 低通滤波器　　B. 带通滤波器
   C. 带阻滤波器　　D. 高通滤波器
2. 如果要设计一个滤波器是的信号通过时去除细节部分，请问用（ ）滤波器。
   A. 低通滤波器　　B. 带通滤波器
   C. 带阻滤波器　　D. 高通滤波器
3. 将带有椒盐噪音的图像进行去噪，应使用（ ）滤波器。
   A. 高通　　B. 维纳
   C. 均值　　D. 低通

二、思考题

1. 对于滤波器和检测器来说，为什么滤波是比检测更难进行。
2. 用维纳滤波器分别对含有高斯噪声和椒盐噪声的图像进行去噪，比较并讨论去噪效果。

三、上机题

1. 编写一个程序，向一幅图像加入椒盐噪声，然后分别进行高斯低通滤波、sobel 滤波、prewitt

滤波、拉普拉斯滤波、高斯拉普拉斯滤波、均值滤波、模糊增强滤波、高斯高通滤波，对比得到的结果。

2. 编写一个程序，向一幅图像分别加入高斯噪声和椒盐噪声，在分别用中值滤波器滤波，观察得到的结果。

3. 编写一个程序，向一幅图像分别加入高斯噪声和椒盐噪声，在分别用状态统计滤波器滤波器滤波，观察得到的结果。

第 8 章　小波分析

近年来人们对一些新的变换技术产生了相当大的兴趣，这些新技术专门是面向图像压缩、边缘和特征检测以及纹理分析的。这些技术包括多分辨率分析、时频域分析、金字塔算法和小波变换等。本章将回顾传统傅立叶变换和其他类似变换的一些局限性，并定义三类在某些应用领域中可以改进性能的小波变换。

这里将变换限制在实值、可测且平方可积的一维和二维函数范围中。之所以是一维或二维是因为由它们所组成的信号和图像正是人们感兴趣的。跟以前一样，为简单起见这里在一维情况下介绍每个概念，然后再把它推广到二维，以便应用于图像。

本章的基本内容如下：

（1）引言。

（2）傅立叶变换的局限性。

（3）连续 Wavelet 变换。

（4）离散小波变换。

（5）多分辨率分析。

（6）二维多分辨率分析。

（7）小波变换的应用。

8.1　引言

任何一个事物，从不同的角度观察，都对应着多个描述空间（这里定义描述空间为 C），每个描述空间 C 都由自身的特征描述基构成，若这些特征基可以描述出 C 中不同事物（或者说描述空间的元素可以被特征基表示），则称特征基在 C 中是完备的。若这些特征基两两之间不相关，则称其正交。当然完备并不要求正交，但是正交的好处在于每个特征基上描述的信息和其他特征基不相关，从而消除了信息的冗余部分。

描述空间也称为描述域。不同特征基也有不同描述和运算规则。所以可以将一个信号（例如图像信息）在描述域 A 中的特征转化为描述域 B 中的特征，从而让信号更符合人们的观察或认知的角度。传统的傅立叶变换就是引入了无穷余弦基和正弦基来无穷逼近 L_2 空间中的函数。因为余弦基和正弦基的许多优秀性质，所以傅立叶变换被广泛地应用。在图像压缩中，人们就是利用了图像数据的特性，将其转化为符合其特性描述的空间上，从而更好地描述图像而达到压缩的目的。而自然图像的数据特征就是其中相邻的像素点的颜色在一个大的概率上相关，否则看到的就是一片颜色乱变的点。对此人们引入图像的频域的概念，对于大范围内平缓变化的信息，叫做低频信息，对于小范围内变化很快的信息，称其为高频信息，并将这些信息对应频域上的数值。离散傅立叶变换就是这样的一种变换，它以变化平缓的波形来描述低频信息，以变化快速的波形来描述高频信息，再结合其他的特征，进行压缩。

但是傅立叶变换存在明显的一些不足。例如，当如果要取得较好的低频信息，则需要相对较长的变换窗口。如果要取得较好的高频信息，则需要较短的窗口。这样就引起了矛

盾。小波变换则应运而生，为了解决傅立叶变换的不足，haar 小波变换就是一个很好的应用的例子。

这里以图像来说明建立空间特征基和小波变换的关系。假设有一幅图像，从不同分辨率来考察：当离得很远来看，可能会把 128 个像素点看成一点，记此时构成的描述空间为 V_0；再走近一些，把 64 个像素点看成一点，记此时构成描述空间 V_1；再走近一些，把 16 个像素点看成一点，记此时构成描述空间 V_2，依此类推。那么，凡是 V_i 空间内可以描述的图像，V_{i+1} 空间内皆可以描述，并且描述得更细致，所以有这样的关系：$V_i \subset V_{i+1}$。

这里记 $V_{i+1} = V_i + W_i$，即 V_i 和 W_i 构成 V_{i+1} 空间。即：

$$V_{i+1} = V_i + W_i = V_{i-1} + W_{i-1} + W_i = \cdots\cdots$$

若 $V_i \perp W_i$，则称 V_i 为 W_i 的正交补空间，实际中并不一定要求正交。若记 P_i 为图像在 V_i 空间的描述，则：

$$D_i = P_{i+1} - P_i$$

就表示了图像在这两个描述空间的细节差异，因为 $V_{i+1} = V_i + W_i$，所以 D_i 为图像在 W_i 空间上的描述。即 W_i 空间描述了细节差异。如果 $W_i \perp W_j$，并且在 W_i 空间中能找到一组标准正交基，其基本函数（小波函数一般是有基本函数平移产生的）是高（带）通的，就称其为小波函数。而 $W_i \perp W_j$，可以理解为不同分辨率下的细节差异不相关，从而消除冗余。由前面的介绍可知 V_3 有如下的表示：

$$V_3 = W_2 + W_1 + W_0 + V_0$$

相应的得到：

$$P_3 = D_2 + D_1 + D_0 + P_0$$

即最清晰分辨率下的图像可以由不同分辨率下的细节差异和最高分辨率下的图像合成得到。由概率特性知道细节差异在大范围内是一个较小的值。如果引用频域的概念来看，低频信息就是 P_0，高频信息就是 D_i，这里的低频和高频就和傅立叶有所不同。而从分析中，可以知道随着频率的不同，其数字对应的空间窗口大小也不同，这正好满足人们的要求。而剩下的任务就是如何构造 W_i，构造过程涉及到 Mallat 塔式算法，详细内容可以查阅相关资料。

在数学上，小波是傅立叶分析理论的一个分支。两者的共同点是：都寻求把复杂现象通过分解成数学上简单的成分来理解，并且都取得了极大的成功。不同点在于：小波分析不用无限延展的正、余弦波，而是依赖于对一个适当选择的母波进行平移和伸缩，而这个母波甚至可以集中在一个有限的区间中。这使得小波分析比傅立叶分析更灵活。小波分析有着广泛的应用前景。

8.2 傅立叶变换的局限性

传统的傅立叶变换有以下两种形式：

$$F(s) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi st} dt$$

$$f(t) = \int_{-\infty}^{\infty} F(s) e^{j2\pi st} ds$$

或：

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt = \int_{-\infty}^{\infty} f(t) \overline{e^{+j\omega t}} dt$$

$$f(t)=\frac{1}{2\pi}\int_{-\infty}^{\infty}F(\omega)e^{j\omega t}d\omega$$

由变换核决定变换。

以下介绍加窗傅立叶变换：

这里考虑非平稳信号：频域特性随时间变化而变化，如音乐信号、语言信号、探地信号、边缘。对这类信号用傅立叶分析不能提供完全的信号。这里知道信号的频率，但不知频率信号到底出现在哪个时间段。并且空间和频率域不能同时局部化。所以，在1946年，Gabor提出加窗傅立叶变换。

由于$\xi\left\{e^{-at^2}\right\}=\int_{-\infty}^{\infty}e^{-ax^2}e^{-j\omega x}dx=\sqrt{\frac{\pi}{a}}e^{-\frac{\omega^2}{4a}}$，所以设：

$$g(t)=e^{-\frac{t^2}{2}} \tag{8.2.1}$$

则：

$$G(\omega)=\sqrt{2\pi}e^{-\frac{\omega^2}{2}} \tag{8.2.2}$$

加窗傅立叶变换如下：

$$\begin{aligned}G_f(\omega,\tau)&=\int_{-\infty}^{\infty}f(t)g(t-\tau)e^{-j\omega t}dt\\&=\int_{-\infty}^{\infty}f(t)\overline{g_{\omega,\tau}(t)}dt\end{aligned} \tag{8.2.3}$$

其中：

$$g_{\omega,\tau}(t)=g(t-\tau)e^{j\omega t} \tag{8.2.4}$$

重建公式（即逆变换）为：

$$\begin{aligned}f(t)&=\frac{1}{2\pi}\int_{-\infty}^{\infty}d\omega\int_{-\infty}^{\infty}G_f(\omega,\tau)g_{\omega,\tau}(t)d\tau\\&=\frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}G_f(\omega,\tau)g_{\omega,\tau}(t)d\omega d\tau\end{aligned} \tag{8.2.5}$$

$G_f(\omega,\tau)$称为加窗傅立叶变换，或称为短时傅立叶变换，或称为STFT。时频同时局部化。

其意义在于$G_f(\omega,\tau)$是$f(t)\bullet g(t-\tau)$的傅氏变换，可以写为：

$$G_f(\omega,\tau)=F(\omega)*G(\omega)e^{jw\tau} \tag{8.2.6}$$

即$G_f(\omega,\tau_0)$实际上是$F(\omega)$经$G(\omega)e^{jw\tau_0}$平滑后的结果。亦即窗口函数 g(t)的频域窗口$2\Delta\omega$的大小决定了Gabor变换的频域分辨率。

但是这样的变换有两个缺点：

（1）时频分辨率固定$\Delta t=1$，$\Delta\omega=1$，无论对什么信号都一样。

（2）$g_{\omega,\tau(t)}$，无论如何离散化，ω、τ都不能获得正交基，因为它无振荡性。

所以为了提高时频分辨率，如果希望Δt、$\Delta\omega$都无限小，但这是不可能。因为：

（1）由相似性定理Δt、$\Delta\omega$不可能同时都小。

（2）由海森堡准则知道：

$$\Delta t\bullet\Delta\omega\geqslant c\quad（c为常数） \tag{8.2.7}$$

下面寻求改进的方法：寻找一个变换核 $\psi_a(t)$，使得：

（1）Δt 和 $\Delta\omega$ 能随 a 伸缩。

（2）$\psi_a(t)$ 离散化后能成为正交基。

8.3 连续 Wavelet 变换

上节中的变换核函数 $\psi(t)$ 通常要求满足以下三个条件：

（1）窗口性：紧支性，即存在 C>0,当 $|t|$>C 时，$\psi(t)=0$。

（2）正交性：振荡性。设 $\psi(t)$ 是平方可积函数，若其傅立叶变换 $\hat{\psi}(\omega)$ 满足：

$$C_\psi = \int_R \frac{|\hat{\psi}(\omega)|^2}{\omega} d\omega < \infty \tag{8.3.1}$$

则称 $\psi(t)$ 为一个基本小波或小波母函数。式（8.3.1）为小波函数的可容许条件（或重构条件）。

所以由式（8.3.1）可以得到：

$$\hat{\psi}(\omega)\big|_{\omega=0} = 0 \Rightarrow \int_R \psi(t)dt = 0 \tag{8.3.2}$$

（3）带通性：$\hat{\psi}(\omega)\big|_{\omega=0}=0$。将 $\psi(t)$ 以尺度因子 a，平移因子 τ 进行平移伸缩：

$$\psi_{a,\tau}(t) = a^{-\frac{1}{2}}\psi\left(\frac{t-\tau}{a}\right) \qquad a>0, \tau \in R \tag{8.3.3}$$

下面来看小波变换：

$$WT_f(a,\tau) = \frac{1}{\sqrt{a}}\int_R f(t)\overline{\psi\left(\frac{t-\tau}{a}\right)}dt \tag{8.3.4}$$

重构：

$$f(t) = \frac{1}{c_\psi}\int_0^{+\infty}\frac{da}{a^2}\int_{-\infty}^{\infty} WT_f(a,\tau)\frac{1}{\sqrt{a}}\psi\left(\frac{t-\tau}{a}\right)d\tau \tag{8.3.5}$$

其中：$C_\psi = \int_0^{+\infty}\frac{|\hat{\psi}(\omega)|^2}{\omega}d\omega < \infty$。

小波母函数通常是一个带通滤波器或高频增强滤波器。

例如高斯差分函数（DOG 小波）：

$$\psi(t) = e^{-t^2/2} - \frac{1}{2}e^{-t^2/8} \tag{8.3.6}$$

一般连续可微的紧支撑函数的导数都是基本小波。

例如：

Gauss 函数：

$$\phi(t) = e^{-\frac{t^2}{2}}$$

$$\psi(t) = \phi'(t) = -te^{-\frac{t^2}{2}} \tag{8.3.7}$$

下面介绍窗口宽度：小波的分辨率。

设小波母函数 $\psi(t)$ 窗口中心为 t_0，大小为 Δt，$\hat{\psi}(\omega)$ 窗口中心为 ω_0，大小为 $\Delta\omega$。$\psi_{a,\tau}(t)$ 的窗口中心和大小改变如下：

$t_{a,\tau} = at_0 + \tau \qquad \Delta t_{a,\tau} = a\Delta t$

$\omega_{a,\tau} = \frac{1}{a}\omega_0 \qquad \Delta\omega_{a,\tau} = \frac{1}{a}\Delta\omega$

窗口面积：$\Delta\omega_{a,\tau} \bullet \Delta t_{a,\tau} = \Delta t \bullet \Delta\omega \geqslant \frac{1}{2}$，但由 $\omega_{a,\tau} = \frac{1}{a}\omega_0, \Delta\omega_{a,\tau} = \frac{1}{a}\Delta\omega, \Delta\omega_{a,\tau}$ 随 $\omega_{a,\tau}$ 大而大，小而小。

小波变换的滤波器特性，即式（8.3.4）：

$$WT_f(a,\tau) = \frac{1}{\sqrt{a}}\int_R f(t)\overline{\psi(\frac{t-\tau}{a})}dt$$

是内积形式。但是，若令 $\tilde{\psi}_a(t) = \psi_a^*(-t)$，则小波变换可以写为：

$$WT_f(a,\tau) = f * \tilde{\psi}_a(t)$$

注意：

① a 是可变的，因而小波变换是对 f(t)用一组（线性）卷积滤波器进行滤波。

② a 的每个值定义一个滤波器，a 越大，滤波器的频率越低。

尺度及位移均连续变化的连续小波基是一种过度完全基。连续小波变换后总比信号多一个自变量，是超完备的。变换后的系数间存在很大的有冗余度。

8.4 离散小波变换

对连续小波变换核 $\psi_{a,\tau}(t) = a^{-\frac{1}{2}}\psi(\frac{t-\tau}{a})$，令：

$a = a_0^m$，$a_0 \neq 1$（一般为 $a_0 = 2$），$\tau = a_0^m \bullet nT_s$

（采样间隔 τ 满足 Nyquist 采样定理，并且把 t 轴用 T_s 归一化），则：

$$\psi_{m,n}(t) = \psi_{a,\tau}(t) = 2^{-\frac{m}{2}}\psi(\frac{t-2^m n}{2^m}) = 2^{-\frac{m}{2}}\psi(2^{-m}t-n) \tag{8.4.1}$$

m 为尺度因子，随 m 小，$\psi_{m,n}$ 定义域也小，平移间隔也小，能表达更细致的信号特性。

离散小波变换：

$$WT_f(m,n) = \int_R f(t)\bullet\overline{\psi_{mn}(t)}dt \tag{8.4.2}$$

当{ $\psi_{m,n}(t)$ }为正交基：

$$f(t) = \sum_{m,n} < f,\psi_{m,n} > \psi_{m,n} \tag{8.4.3}$$

正交小波基：haar 小波（1910 年）。

$$h(t) = \begin{cases} 1 \quad , & 0 \leqslant t < \frac{1}{2} \\ -1 \quad , & 2 < t \leqslant 1 \\ 0 \quad , & \text{其他} \end{cases} \tag{8.4.4}$$

$$h_{m,n}(t) = 2^{-\frac{m}{2}}h(2^{-m}t-n)\text{，}(m,n \in z) \tag{8.4.5}$$

还有更多的正交小波基吗?

由此引出了 Daubechies 正交小波滤波器系数，但是没有函数表达式。接下来将讲述多分辨率分析。

8.5 多分辨率分析

8.5.1 多分辨率分析

尺度可理解为相机的镜头，尺度大，看物体轮廓；尺度小，看细节。

多分辨率分析的思想是：随着尺度由大到小的变化，在各个尺度上可以由粗到精地观察目标。

定义：

设$\{V_m\}_{m\in Z}$是$L^2(R)$的一列子空间，其中，Z 为全体整数集，R 为全体实数集，满足下列性质：

（1）一致单调性：$V_m \subset V_{m-1}$对所有的 $m\in Z$ 成立。

（2）渐进完全性： $\cap V_m = \{0\}$，$\cup V_m = L^2(R)$。

（3）伸缩规则性：$f(x)\in V_m \Leftrightarrow f(2x)\in V_{m-1}$。

（4）平移不变性：$f(t)\in V_0 \Rightarrow f(t-n)\in V_0$，对所有 $n\in Z$。

（5）正交基存在性：存在函数$\varphi(x)\in V_0$，使得：

$$\{\varphi_{m,n}(x)=2^{-\frac{m}{2}}\varphi(2^{-m}x-n)\}_{n\in Z} \tag{8.5.1}$$

构成 V_m的标准正交基，$m\in Z$。就称$\{V_m\}_{m\in Z}$是$L^2(R)$的一个多分辨率分析，并称$\varphi(x)$为尺度函数。

设 W_m为 V_m的正交补空间，即：

$$V_{m-1}=V_m\oplus W_m,\qquad W_m\perp V_m \tag{8.5.2}$$

W_m称为小波空间或细节空间。

设$\{\psi_{0,n}(x)\}_{n\in Z}$为空间 W_0的一组正交基，则$\{\psi_{m,n}(x)=2^{-\frac{m}{2}}\psi(2^{-m}x-n)\}_{n\in Z}$为 W_m的正交基。

由多分辨率分析的定义：

$$V_0=V_1\oplus W_1=V_2\oplus W_2\oplus W_1=V_3\oplus W_3\oplus W_2\oplus W_1=\ldots \tag{8.5.3}$$

$$L^2(R)=\oplus^{J}_{j\in J}W_j\oplus V_J \tag{8.5.4}$$

$$L^2(R)=\oplus_{j\in Z}W_j \tag{8.5.5}$$

信号的多分辨率分析：

设 $f_c^m(t)$ 为 f(t)在 V_m中投影得到的 m 尺度下的“粗”信号，则：

$$f_c^m(t)=\sum_n c_{m,n}\varphi_{m,n}(t) \tag{8.5.6}$$

其中，$c_{m,n}=\langle f(t),\varphi_{m,n}(t)\rangle$称为尺度展开系数。

同理：

$$f_d^m(t)=\sum_n d_{m,n}\psi_{m,n}(t) \tag{8.5.7}$$

其中，$d_{m,n}=\langle f(t),\psi_{m,n}(t)\rangle$称为小波展开系数。

根据式（8.5.6）和式（8.5.7），f(t)可以展开为：

$f(t)=\sum_{m=-\infty}^{J}\sum_n d_{m,n}\psi_{m,n}(t)+\sum_n c_{J,n}\varphi_{J,n}(t)$，其中右边第一项是图像细节，第二项是平滑图

像。

这是网络上信号传输的新模式。

双尺度方程：

由于 $V_0 \subset V_{-1}, W_0 \subset V_{-1}$，因此φ(x)和ψ(x)必然属于 V_{-1}，φ(x)和ψ(x)可以用 V_{-1} 空间的正交基 $\varphi_{-1,n}$ (x)线性展开，即：

$$\varphi(x) = \sqrt{2} \sum_{n \in Z} h(n)\varphi(2x-n) \tag{8.5.8}$$

$$\psi(x) = \sqrt{2} \sum_{n \in Z} g(n)\varphi(2x-n) \tag{8.5.9}$$

其中 $g(n) = (-1)^n h(1-n)$。

故式（8.5.8）和（8.5.9）为双尺度差分方程。

通过多分辨率分析可以构造出一对正交镜像滤波器，只需令 H 和 G 分别是冲激响应为 $\{h(n)\}_{n \in Z}$ 和 $\{g(n)\}_{n \in Z}$ 的滤波器即可，即：

$$H(\omega) = \frac{1}{\sqrt{2}} \sum_{n \in Z} h(n) e^{-j\omega n} \tag{8.5.10}$$

$$G(\omega) = \frac{1}{\sqrt{2}} \sum_{n \in Z} g(n) e^{-j\omega n} \tag{8.5.11}$$

可以证明：

$$\sum_{n \in Z} h(n) = \sqrt{2} \tag{8.5.12}$$

$$\sum_{n \in Z} g(n) = 0 \tag{8.5.13}$$

即：

$$H(\omega = 0) = 1 \tag{8.5.14}$$

$$G(\omega = 0) = 0 \tag{8.5.15}$$

故，H 为低通滤波器，G 为高通滤波器。

8.5.2 Mallat 塔式算法

定义 $L^2(R)$ 到 V_m 和 W_m 上的投影算子 P_m 和 Q_m 如下：

$$P_m f(t) = \sum_{k \in z} < f, \varphi_{m,k} > \varphi_{m,k}(t) = \sum_{k \in z} c_{m,k} \varphi_{m,k}(t) \tag{8.5.16}$$

$$Q_m f(t) = \sum_{k \in z} < f, \psi_{m,k} > \psi_{m,k}(t) = \sum_{k \in z} c_{m,k} \psi_{m,k}(t) \tag{8.5.17}$$

$P_m f(x)$，$Q_m f(x)$ 分别是函数 $f(x)$ 在 V_m 和 W_m 上的最佳 L^2 逼近函数。

可以证明：

$$c_k^{m+1} = \sum_n h_{n-2k} c_n^m \tag{8.5.18}$$

$$d_k^{m+1} = \sum_n g_{n-2k} c_n^m \tag{8.5.19}$$

信号 f 一般被认为是分辨率有限的信号，可以假设：

令：

$$f = \sum_n c_n^0 \varphi_{0,n}, \quad f(x) \in V_0 \subset L^2(R)$$

其中 $C_n^0 = < f, \varphi_{0,n} >$，通过式（8.5.18）和（8.5.19）可以计算所有尺度下的小波系数。反过来，由于 $V_m = V_{m+1} \oplus W_{m+1}$，可以证明有如下的重建公式：

$$c_k^m = \sum_n h_{k-2n} C_n^{m+1} + \sum_n g_{k-2n} d_n^{m+1} \quad (8.5.20)$$

写成滤波器的形式为：

$$c^{m+1} = Hc^m = \sum_n h_{n-2k} c_n^m \quad (8.5.21)$$

$$d^{m+1} = Gc^m = \sum_n g_{n-2k} c_n^m \quad (8.5.22)$$

假设 H^* 和 G^* 分别为 H 和 G 的共轭算子，即：

$$(H^* a)_n = \sum_k h_{n-2k} a_k$$

$$(G^* a)_n = \sum_k g_{n-2k} a_k$$

还有：

$$c^m = H^* c^{m+1} + G^* d^{m+1} \quad (8.5.23)$$

H^*、G^*、H、G 还满足等式：

$$H^* \mathrm{G} = \mathrm{H} G^* = 0 \quad (8.5.24)$$

$$H^* \mathrm{H} + G^* \mathrm{G} = \mathrm{I} \quad (8.5.25)$$

其中 I 是恒同算子，H、G 称为正交镜像滤波器（QMF）。

二维多分辨率分析：

二维小波可以由一维小波通过同尺度张量积的方法推广出来。

记：

$$V_m^2 = Span\{F(t_1, t_2) = f(t_1) g(t_2) : f, g \in V_m\} \quad (8.5.26)$$

可以验证 $\{V_m^2\}_{m \in Z}$ 构成一个二维多分辨率分析，尺度函数为 $\Phi(t_1, t_2) = \varphi(t_1)\varphi(t_2)$，并且：

$$V_{m-1}^2 = V_m^2 \oplus (V_m \otimes W_m) \oplus (W_m \otimes V_m) \oplus (W_m \otimes W_m) = V_m^2 \oplus W_m^2$$

若记：$\tilde{W}_m^1 = (V_m \otimes W_m)$，$\tilde{W}_m^2 = (W_m \otimes V_m)$，$\tilde{W}_m^3 = (W_m \otimes W_m)$。则正交补空间 W_m^2 由三个子空间 $\tilde{W}_m^1$，$\tilde{W}_m^2$ 和 $\tilde{W}_m^3$ 的值和组成，它们分别由下列三个小波函数 $\psi^1(x,y) = \varphi(x)\psi(y)$，$\psi^2(x,y) = \psi(x)\varphi(y)$ 和 $\psi^3(x,y) = \psi(x)\psi(y)$ 所构成的正交基生成。

同理可以推出二维信号的小波分解与合成公式如下：

给定平方可和的二维离散信号 $\{x_{m,n}\}_{m,n \in Z} \in l^2$，令：

$$c_{0,m,n} = x_{m,n} \qquad m, n \in Z$$

$\{c_{0,m,n}\}_{m,n \in Z}$ 可以看作是某个函数 f(x,y)在 V_0^2 上的展开系数。

假设 $c_j = \{c_{j,m,n}\}_{m,n \in Z}$，$d_j^1 = \{d_{j,m,n}^1\}_{m,n \in Z}$，$d_j^2 = \{d_{j,m,n}^2\}_{m,n \in Z}$ 和 $d_j^3 = \{d_{j,m,n}^3\}_{m,n \in Z}$ 分别是某个函数 f(x,y)在 V_m^2，$\tilde{W}_m^1$，$\tilde{W}_m^2$ 和 $\tilde{W}_m^3$ 上的展开系数，则二维图像的小波分解递推公式（Mallat 算法）为：

$$c_{j+1,m,n} = (\mathrm{H} \otimes \mathrm{H})(c_j)_{m,n} = \sum_{k,l} c_{j,k,l} h_{k-2m} h_{l-2n} \quad (8.5.27)$$

$$d_{j+1,m,n}^1 = (\mathrm{H} \otimes \mathrm{G})(c_j)_{m,n} = \sum_{k,l} c_{j,k,l} h_{k-2m} g_{l-2n} \quad (8.5.28)$$

$$d^2_{j+1,m,n} = (G\otimes H)(c_j)_{m,n} = \sum_{k,l} c_{j,k,l} g_{k-2m} h_{l-2n} \tag{8.5.29}$$

$$d^3_{j+1,m,n} = (G\otimes G)(c_j)_{m,n} = \sum_{k,l} c_{j,k,l} g_{k-2m} g_{l-2n} \tag{8.5.30}$$

合成算法是：

$$c_{j,m,n} = \sum_{l,k} c_{j+1,k,l} h_{m-2k} h_{n-2l} + \sum_{l,k} d^1_{j+1,k,l} h_{m-2k} g_{n-2l}$$

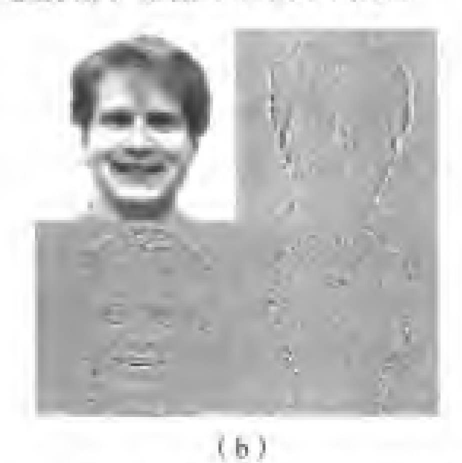
$$+\sum_{l,k} d^2_{j+1,k,l} g_{m-2k} h_{n-2l} + \sum_{l,k} d^3_{j+1,k,l} g_{m-2k} g_{n-2l} \tag{8.5.31}$$

8.5.3　图像的小波特性

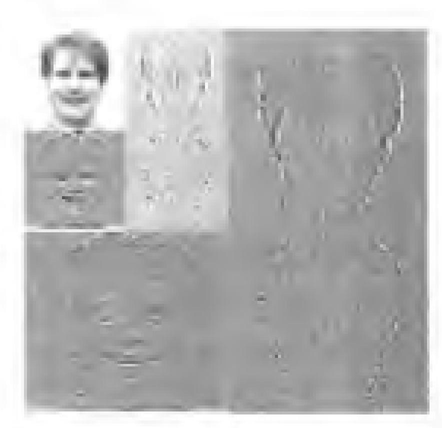
图像 I（如图 8-1（a）所示）经过一层小波分解后，将会得到如图 8-1（b）所示的结果。同理，可进行多层小波分解，如图 8-1（c）所示。



（a）



（b）



（c）

图 8-1　图像信号小波分解示意图

（a）原图像。

（b）一层二维小波分解。

（c）二层二维小波分解。

二维小波可以用同尺度张量积方法由一维小波推广而来，二维小波变换在图像处理中有广泛的应用，在图像处理中通常使用如下与二维小波变换等价的滤波器运算：

给出平方可和的二维离散图像信号 $\{s_{m,n}\}_{m,n\in Z}$，令：

$c_{0,m,n} = s_{m,n}$，$m,n\in Z$

则二维图像的小波分解递推公式为：

$$c_{j,m,n} = (H\otimes H)(c_{j+1})_{m,n} = \sum_{k,l} c_{j+1,k,l} h_{2m-k} h_{2n-l} \tag{8.5.32}$$

$$d^1_{j,m,n} = (H\otimes G)(c_{j+1})_{m,n} = \sum_{k,l} c_{j+1,k,l} h_{2m-k} g_{2n-l} \tag{8.5.33}$$

$$d^2_{j,m,n} = (G\otimes H)(c_{j+1})_{m,n} = \sum_{k,l} c_{j+1,k,l} g_{2m-k} h_{2n-l} \tag{8.5.34}$$

$$d^3_{j,m,n} = (G\otimes G)(c_{j+1})_{m,n} = \sum_{k,l} c_{j+1,k,l} g_{2m-k} g_{2n-l} \tag{8.5.35}$$

只有对于具有紧支集的正交小波基，其脉冲响应{h(n)}、{g(n)}才为有限项不为零，从而才能实现上述离散小波变换的分解。

图像 I（如图 8-1（a）所示）经过一层小波分解后，将会得到如图 8-2（b）所示的结

果。同理，可进行多层小波分解。

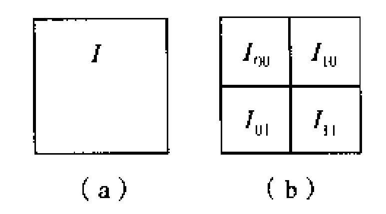


图 8-2 图像信号正交小波分解示意图

（a）原图像。

（b）一层小波分解。

其中子图像 I_{00} 是低频分量，为原图的平滑像；子图像 I_{01} 是水平方向低频，垂直方向高频的分量，表现原图的水平边缘；子图像 I_{10} 是垂直方向低频，水平方向高频的分量，表现原图的垂直边缘；子图像 I_{11} 是垂直方向和水平方向均为高频的分量，表现原图的斜边缘。

1. 多分辨率特性

一般而言，尺度越大，分辨率越小，图像的尺寸也越小。对于 128×128 的图像，假设其分辨率为 1，做一次二维小波分解后，其分辨率为原来 1/2，图像的尺寸为原来的 1/4（64×64）。

图像在不同分辨率或不同尺度下具有不同的特性。在高分辨率下，图像的特性主要表现在图像的细节特征上，而在低分辨率下，图像的特性主要表现在图像的宏观结构特征上，可以根据不同的需要选择不同分辨来描述图像或选择不同大小的图像来处理。

2. 方向性选择

图像中的边缘、纹理等往往有很强的方向性[13]，图像识别的一个经典方法是用图像在各方向的投影变换值作为图像特征来进行分类识别[15]，识别结果较理想。但由于图像边缘、纹理的方向性体现在图像的局部变化和沿某一方向上的全局性中，传统方法所提取的特征不能体现这一性质，而二维小波变换正好反映这一特性。

二维小波的分解可以理解为图像在一组独立的，具有空间方向性的频道上的分解。其中，子图像 I_{00} 对应低频成分，子图像 I_{01} 代表 0°方向的信息，子图像 I_{10} 代表 90°方向的信息，子图像 I_{11} 代表 45°或 135°方向的信息。

3. 局部分析特征

小波变换由于在空间和频域上具有良好的局部特性，可以将空间和频域统一于一体来研究图像信号。Wickerhauser[14]指出通过选择一序列镜像滤波器 H、G 的排列叠置，可以获得任何指定频带的局部图像。二维小波变换的局部分析特性也表现为一种提取图像边缘的能力。在图像的二维小波分解中，子图像 I_{01} 给出原图水平边缘信息；子图像 I_{10} 给出原图垂直边缘信息；子图像 I_{11} 给出原图的斜边缘信息。

因此在具体应用中，可以根据具体要求，提取指定频带的局部图像或不同的图像边缘信息。

4. 视觉特性

一般而言，在图像的二维小波分解中，水平边缘信息和垂直边缘信息在视觉上对图像的主观质量影响较大，而斜边缘信息的影响则相对较小。

对于图像识别而言，斜边缘信息最重要，因为它代表图像的宏观结构特征；对于正面

人脸识别而言，水平边缘信息较垂直边缘信息重要，因为眼睛和嘴巴的水平特征比垂直特征明显；对于侧面人脸识别而言，垂直边缘信息较水平边缘信息重要，因为人脸轮廓和鼻子的垂直特征比水平特征明显。

5. 紧支集特性

小波函数的紧支集特性决定了小波变换的计算复杂度。显然，支集越小，滤波器响应序列越短，小波变换的计算复杂度越低，便于实现。Daubechies[2]构造了一系列具有紧支集的正交小波基，在实践中具有广泛的应用。

6. 线性相位特性

Wickerhauser[14]已经证明，采用具有线性相位的滤波器进行图像处理，可以消除图像在边缘处的失真。而且线性相位能保证变换后的图像重心线性偏移，便于边缘检测中的精确定位。

然而，有紧支集的正交小波基除 Haar 系以外不具有线性相位。为了保证线性相位，通常的做法是放弃正交性而采用双正交小波基[4]。

7. 消失矩特性

如果小波函数$\psi(x)$满足：

$\int_{-\infty}^{+\infty} x^m \psi(x)dx = 0$， m = 0,1⋯,k-1。

一般地说，k 越大，小波变换后能量越集中于低频分量I_{00}中，则在子图像I_{01}，I_{10}和I_{11}中会出现更多的零。多分辨分析可以找到两个函数：低通滤波器、带通滤波器。

下面来讨论尺度函数$\phi(t)$和小波函数$\psi(t)$。

$\{\ \phi_{j,k}(t), k\in z\ \}$是正交基，$\phi_{j,k}(t) = 2^{-j/2}\phi(2^j t-k)$ （8.5.36）

$\{\ \psi_{j,k}(t), j,k\in z\ \}$是$L^2(R)$正交基，$\psi_{j,k}(t) = 2^{-j/2}\psi(2^j t-k)$ （8.5.37）

对$\forall J$,和$f(t)$

$$f(t) = \sum_{j=-\infty}^{J}\sum_{k=-\infty}^{\infty} d_{j,k}\psi_{j,k}(t) + \sum_{k=-\infty}^{\infty} C_{Jk}\phi_{J,k}(t) \tag{8.5.38}$$

细节　　　　　平滑像

尺度函数$\phi(t) \Leftrightarrow \{h_0(k)\}$，是低通滤波器。小波函数$\psi(t),\phi(t) \Leftrightarrow \{h_1(k)\}$，是带通滤波器。

如何由$\phi(t) \Rightarrow \psi(t)$呢？步骤如下：

（1）$\{h_0(k)\} \Rightarrow \phi(t)$

设$\sum_k h_0(k) = \sqrt{2}$，且：

$$\sum_k h_0(k)h_0(k+2l) = \delta(l) \tag{8.5.39}$$

则存在一个尺度函数：

$$\phi(t) = \sqrt{2}\sum_k h_0(k)\phi(2t-k) \tag{8.5.40}$$

$$\phi(t) = \lim_{i\to\infty}\eta_i(t) \qquad \text{阶递函数逼近} \tag{8.5.41}$$

其中：

$$\eta_i = \sqrt{2}\sum_n h_0(n)\eta_{i-1}(2t-n) \tag{8.5.42}$$

$$\eta_0(x)=\prod(x) \tag{8.5.43}$$

（2） $\phi(t)\Rightarrow\{h_0(k)\}$

$$<\phi(t-m),\phi(t-n)>=\delta_{m,n} \tag{8.5.44}$$

$$h_0(k)=<\phi_{1,0}(t),\phi_{0,k}(t)>$$

$$\phi_{1,0}(t)=\frac{1}{\sqrt{2}}\phi(\frac{t}{2})=\sum_k h_0(k)\phi(\frac{2t}{2}-k)=\sum_k h_0(k)\phi(t-k)$$

$$\therefore h_0(k)=<\phi_{1,0}(t),\phi_{0,k}(t)>$$

（3） $\phi(t)\Rightarrow\psi(t)$

$$h_1(k)=(-1)^n h_0(-k+1)$$

$$\psi(t)=\sqrt{2}\sum_k h_1(k)\phi(2t-k)$$

8.6 二维小波变换

二维小波由一维小波同尺度张量积而成，二维尺度 $\phi(x,y)=\varphi_{(x)}\phi_{(y)}$。

三个二维基本小波：

$$\varphi^1(x,y)=\varphi_{(x)}\varphi_{(y)} \qquad \varphi^2(x,y)=\varphi_{(x)}\phi_{(y)} \qquad \varphi^3(x,y)=\varphi_{(x)}\varphi_{(y)}$$

$$\left\{\varphi^l_{j,m,n}(x,y)\right\}=\left\{2^j\varphi^l(2^jx-2m,2^jy-n)\right\},j\geqslant 0,l=1,2,3$$

为 $l^2(R^2)F$ 的正交基归一基。

对于图像 $f(x,y)$，二维变换：

$$\alpha^j_{m,n}=\iint_{R^2}f(x,y)\overline{\varphi_{j,m}}(x)\overline{\phi_{j,n}}(y)dxdy$$

$$\beta^j_{m,n}=\iint_{R^2}f(x,y)\overline{\phi_{j,m}}(x)\overline{\varphi_{j,n}}(y)dxdy$$

$$\gamma^j_{m,n}=\iint_{R^2}f(x,y)\overline{\varphi_{j,m}}(x)\varphi_{j,n}(y)dxdy$$

$$S^j_{m,n}=\iint_{R^2}f(x,y)\overline{\phi_{j,m}}(x)\overline{\phi_{j,n}}(y)dxdy$$

$$f(x,y)=\sum_{j=1}^{J}\sum_{m,n}[\alpha^j_{m,n}\varphi_{j,m}(x)\phi_{j,n}(y)+\beta^j_{m,n}\phi_{j,m}(x)\varphi_{j,n}(y)$$
$$+\gamma^j_{m,n}\varphi_{j,m}(x)\varphi_{j,n}(y)]+\sum_{m,n}S^J_{m,n}\phi_{J,m}(x)\phi_{J,n}(y)$$

对于离散化的图像 $S^0_{i,l},h_0,h_1$ 分别为低通和高通滤波器、维也纳小波变换的快速分解公式如下：

$$\alpha^j_{i,l}=\sum_{k,n}h_1(k-2j)h_0(m-2l)S^{j-1}_{k,m}$$

$$\beta^j_{i,l}=\sum_{k,n}h_0(k-2j)h_1(m-2l)S^{j-1}_{k,m}$$

$$\gamma^j_{i,l}=\sum_{k,n}h_1(k-2j)h_0(m-2l)S^{j-1}_{k,m}$$

$$S^j_{i,l}=\sum_{k,n}h_0(k-2j)h_0(m-2l)S^{j-1}_{k,m}$$

其重构公式为：

$$S_{k,m}^{j-1}=\sum_{i,l}S_{i,l}^{j}h_0(k-2i)h_0(m-2l)+\sum_{i,l}\alpha_{i,l}^{j}h_1(k-2i)$$

$$h_0(m-2l)+=\sum_{i,l}\beta_{i,l}^{j}h_0(k-2i)h_1(m-2l)+\sum_{i,l}\gamma_{i,l}^{j}h_1(k-2j)h_1(9m-2l)$$

8.7 小波变换的应用

1. 图像压缩

对 $\alpha_{i,l}^{j}$、$\beta_{i,l}^{j}$、$\gamma_{i,l}^{j}$ 保留大的，去掉小的，可由重构公式近似重构。离散小波变化把图像分解为一组越来越小的正交归一图像。除此之外，虽然原始图像的灰度直方图可以是任何形状，但是它们的小波变换图像却通常都是单峰并且对称于零的，这就简化了统计特征的分析。

经常遇到的情况是，既可以用少量等级量化的方法，甚至可以完全去除那些系数值较小的方法。Mallat 和其他一些学者已经研究过只从一幅图像小波变换零交叉的位置来重建图像的可能性。虽然完美重建是不可能的，但是许多图像经过很高的压缩仍然有足够的近似程度。

2. 图像融合

对于两幅图像：

$S_{i,l}^{j,1}$	$\alpha_{i,l}^{j,1}$	$\beta_{i,l}^{j,1}$	$\gamma_{i,l}^{j,1}$
$S_{i,l}^{j,2}$	$\alpha_{i,l}^{j,2}$	$\beta_{i,l}^{j,2}$	$\gamma_{i,l}^{j,2}$
$S_{i,l}^{j,3}$	$\alpha_{i,l}^{j,3}$	$\beta_{i,l}^{j,3}$	$\gamma_{i,l}^{j,3}$

取最大的 u，再作逆变换得到融合图像。图像融合将同一对象的两个或更多的图像合成到一幅图像中，以便它比任何原来的一幅图像都更容易为人们所理解。这一技术可以应用于多频谱图像理解以及医学图像处理等领域中，在这些场合，同一物体部件的图像往往是使用不同的成像机理采集得到的。

3. 图像增强

让某些 α、β、γ 增大，再作逆变换，增强图像。DWT 将一幅图像分解为大小、位置和方向都不同的分量。与傅立叶频域里的线性滤波一样，可以在做逆变换之前改变小波变换域中某些系数的幅度。这样就能够有选择地提升所感兴趣地分量而牺牲不需要的东西。

小结

本章主要介绍了傅立叶变换的局限性、连续 Wavelet 变换、离散小波变换、多分辨率分析、二维小波变换、小波变换的应用等内容。

重点：本章的重点主要有以下几个方面：

（1）一个基本小波是一个当 $|x|\to\infty$ 时趋于零的振荡函数。它的频谱与带通滤波器的传递函数相似。

（2）小波变换的一组基函数可由一个基本小波的伸缩和平移而得到。

（3）连续小波变换将一个信号表示为一个双变量的函数：时间和尺度。而将一幅图像表示为一个三变量的函数：两个空间位置和一个尺度。

（4）小波级数展开将一个周期或有限长信号表示为一个系数序列。

（5）离散小波变换将一个有 N 个点的信号表示为 N 个系数。而用 N^2 个系数表示一幅 N×N 的图像。

（6）Haar 变换是最简单的离散小波变换。

（7）DWT 可以直接或间接地用快速小波变换（FWT 或鱼谷型）算法来实施。

（8）可分离二维 DWT 也能够用 FWT 算法实现。

（9）双正交小波系统允许 DWT 使用限制较少（例如对称性）地紧支集小波。

难点：本章难点在于如何理解小波对图像处理的重大意义，如何掌握小波变换的应用原理以及应用领域。

疑点：小波变换是否可以完全代替傅立叶变换。

练习八

一、选择题

1. 导致离散小波变换发展起来的三个技术是（　　）。

   A. 滤波器族理论　　B. 多分辨率或时频尺度分析

   C. 子带编码　　D. 傅立叶变换

2. 和傅立叶变换那样，在小波变换中同样存在三种可能性（　　）。

   A. 连续小波变换　　B. 小波级数展开

   C. 快速小波变换　　D. 离散小波变换

二、思考题

1. 简述小波在图像方面的应用。
2. 小波变换可以应用于降低人脸识别过程中表情对识别率的影响，为什么？

三、上机题

编写一个程序，将一幅图像进行两次小波变换，然后分别去除垂直分量、水平分量和点分量，再用逆变换进行复原，将得到的图像和原图像进行对比。

第9章　图像的灰度直方图及基本运算

通常，在计算机中表示的一幅二维数字图像可表示为一个矩阵，其矩阵中的元素是位于相应坐标位置的图像灰度值，是离散化的整数，一般取 0、1、……、255。这主要是因为计算机中的一个字节所表示的数值范围是 0~255。另外，人眼也只能分辨 32 个左右的灰度级。所以，用一个字节表示灰度即可。

图像直方图描述了一幅图像的灰度级内容，但是，直方图仅能统计某级灰度像素出现的概率，反映不出该像素在图像中的二维坐标。因此，不同的图像有可能具有相同的直方图。通过灰度直方图的形状，能判断该图像的清晰度和黑白对比度。

如果获得一幅图像的直方图效果不理想，可以通过直方图均衡化处理技术作适当修改，即把一幅已知灰度概率分布图像中的像素灰度作某种映射变换，使它变成一幅具有均匀灰度概率分布的新图像，实现使图像清晰的目的。

本章主要内容如下：

（1）灰度直方图。

（2）直方图的用途。

（3）点运算。

（4）点运算和直方图。

（5）点运算的应用。

9.1　灰度直方图

9.1.1　灰度直方图的概念

直方图（Histogram）是多种空间域处理技术的基础，是图像处理中一种十分重要的图像分析工具。直方图操作能有效地用于图像增强、图像压缩与分割。直方图在软件中易于计算，也适用于商用硬件设备，所以，它们成为实时图像处理的一个流行工具。

任何一幅图像的直方图都包含了丰富的信息。图像的灰度直方图是表示一幅图像灰度分布情况的统计特性图表，从数学上来说图像直方图是图像各灰度值统计特性与图像灰度值的函数，它统计一幅图像中各个灰度级出现的次数或概率；从图形上来说，它是一个二维图，横坐标表示图像中各个像素点的灰度级，纵坐标为各个灰度级上图像各个像素点出现的次数或概率。图像的灰度直方图简称直方图。直方图是用一系列宽度相等、高度不等的矩形表示数据分布的图形。矩形的宽度表示数据范围的间隔，矩形的高度表示在给定间隔内的数据频数。人们常用的是频数直方图。

假设一幅图像像素个数为 n，分 L 个灰度级，N_k 代表第 k 个灰度级出现的个数，直方图函数为：

$$H(k) = N_k \qquad (9.1.1)$$

其中，N_k，$k = 0,1,\ldots L-1$。

有时，直方图也用某一灰度值的像素数 N_k 占总的像素数 N 的百分比作为纵坐标。

如图 9-1（a~d），给出了以四个基本灰度特征（正常、高对比度、暗、亮）做出的图像。其右侧则显示了每幅图案的相应的灰度直方图。在直方图中，可以看到，在正常图像中，直方图的灰度级分布比较均匀。在高对比度图像的直方图中，直方图的成分覆盖了灰度级很宽的范围，而且，像素的分布也不太均匀，只有少数垂线比其他垂线高出很多。在暗图像中，直方图组成成分集中于灰度级低的一侧。相反，亮图像的灰度直方图的组成成分集中于灰度级高的一侧。这样，可以得出结论：如果一幅图像其像素占有全部可能的灰度级且分布均匀，则这样的图像是比较好的，而对于不好的图像，就要用到直方图均衡化。

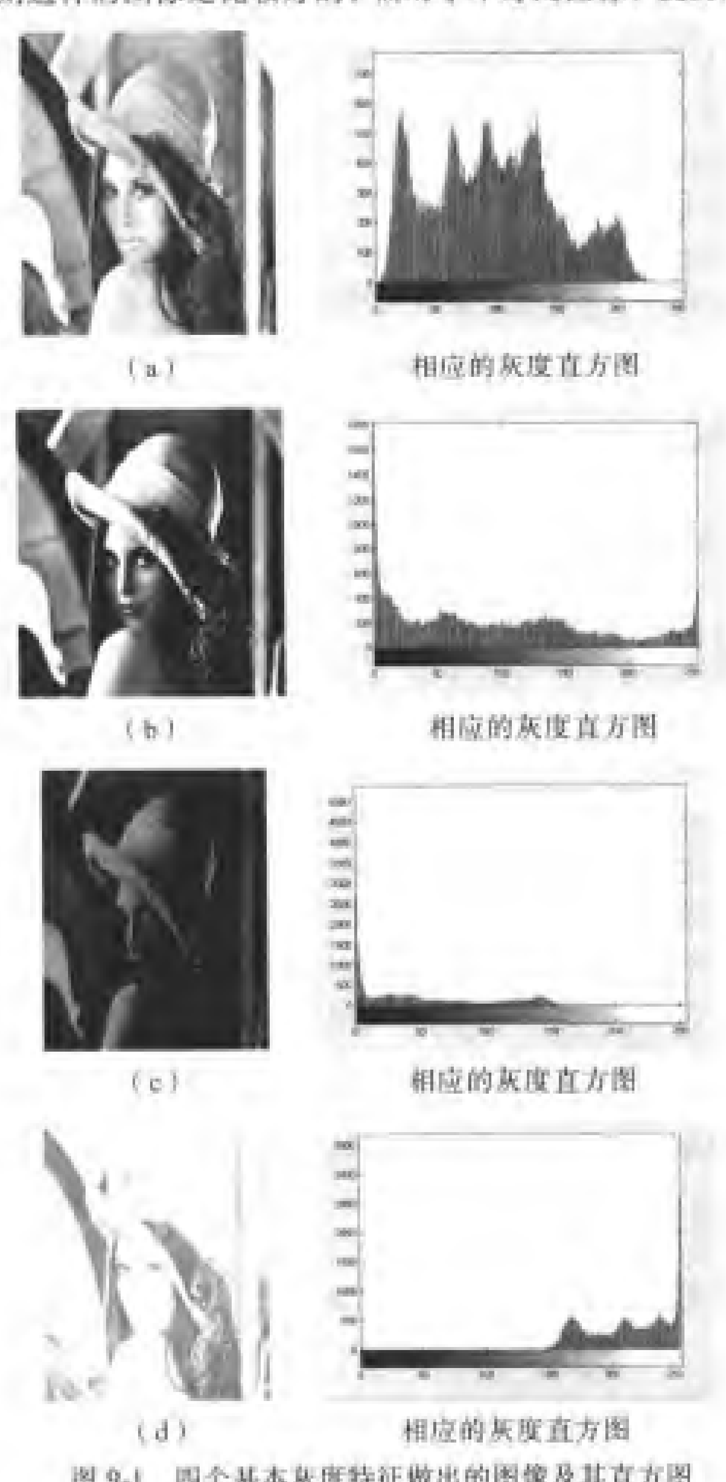


图 9-1 四个基本灰度特征做出的图像及其直方图

9.1.2　二维直方图

人们发现构造高维直方图比一维直方图更为有用。特别是对于研究彩色图像。二维直方图是两个变量：红光图像的灰度值和蓝光图像的灰度值的函数。它统计两个同尺度图像A、B中相同位置的一对像素值的分布关系。

在二维直方图中，坐标（D_A，D_B）的值是指在图像A中具有灰度值D_A，同时在图像B中同一位置具有灰度值D_B的像素对的个数。这是一种多光谱的数字图像，在每个采样点有一个像素，每个像素有多个变量。二维直方图表示像素值在两种灰度级的组合中的分布情况。红-蓝直方图有四个不同的峰值，一个由背景（B）产生，一个由红细胞（R）产生，一个由核（N）产生。

9.1.3　直方图的性质

当一幅图像被描述为直方图后，所有的空间信息都丢失了。直方图描述了每个灰度级具有的像素个数，但无法为这些像素在图像中的位置提供任何线索。即便如此，直方图仍有一些有用的性质：

性质1：一个特定的图像有惟一的直方图，但两幅图像的直方图相同并不能说明图像相同。

性质2：在图像中特定对象的直方图是平移不变的。

性质3：在图像中特定对象的直方图是旋转不变的。

性质4：$\frac{dy}{dx}\int_0^{\infty} H(D)dD =$ 图像的面积。

性质5：对离散图像 $\sum_{i=0}^{255} H(D) = NL \times NS$

其中NL和NS分别为图像的行和列。

性质6：如果一幅图像由两个不连接的区域组成，且每个区域的直方图已知，则整幅图像的直方图是两个区域的直方图之和。显然，该结论可以推广到任何数目的不连接区域的情形。

9.2　直方图的用途

9.2.1　数字化参数

直方图可以作为判断一幅图像是否合理的利用了全部被允许的灰度级范围的指标。一般情况下，一幅图像应该利用全部或几乎全部可能的灰度级，如前页图9-1所示。

如果图像的灰度超出数字化器所能处理的范围，则超出范围的灰度级简单的置为0或255，这样，将在直方图的两端（或一端）产生尖峰。故数字化之前，对直方图进行检查是一个良好的习惯。

9.2.2　图像二值化边界阈值选择

轮廓线可以确立图像中的简单物体的边界，使用轮廓线作为边界的技术被称为阈值化，如一幅图像的前景是浅色的，而背景是深色的，如图9-2所示。则这类物体的直方图

大致如图 9-3 所示。

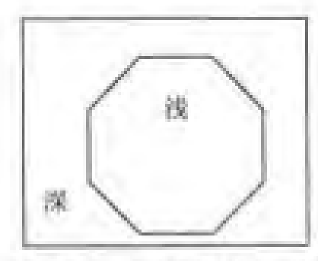


图 9-2 边界明显的图像

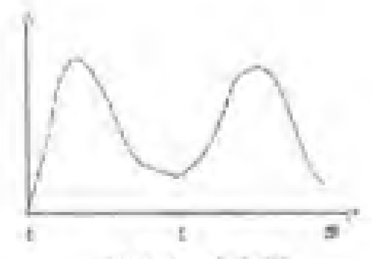


图 9-3 直方图

很显然，深色背景产生直方图的左峰，浅色前景产生直方图的右峰。物体的边界附近具有的灰度级介于两个峰值之间而且像素数目较少，反映在直方图中就是两个峰值之间的谷底。选择谷底 T 使小于 T 的灰度值置为零；大于 T 的灰度值置为 1，则得到二值图像。

也可使小于 T 的灰度值置为零，大于 T 的灰度值不变，得到物体除去背景的图像。如图 9-2 的二值化图像，大致如图 9-4 所示。

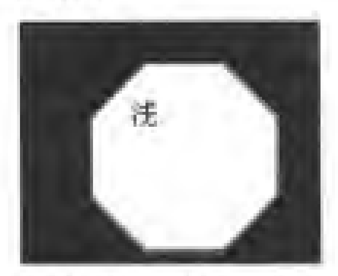


图 9-4 二值化图像

9.2.3 综合光密度

利用综合光密度可以反映图像的质量，其实质量可理解为图像的所有像素的灰度值总和。

设 D(x,y)为连续图像，则综合光密度为：

$$IOD=\int_0^a\int_0^b D(x,y)dxdy \tag{9.2.1}$$

其中，a、b 为所划定的图像区域的边界。

对离散图像，综合光密度为：

$$IOD=\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}D(i,j) \tag{9.2.2}$$

若令 N_k 表示灰度级为 k 的像素的个数，则上式可写为：

$$IOD=\sum_{k=0}^{255}kN_K=\sum_{k=0}^{255}kH(K) \tag{9.2.3}$$

其中，H（K）为灰度级 k 所对应像素个数，为灰度级 K 的函数。显然，上式为一幅图像的所有灰度值加起来，或者说为灰度值加权的直方图之和。

对连续图像，综合光密度为：

$$IOD=\int_0^{\infty}DH(D)dD \tag{9.2.4}$$

由（9.2.1）和（9.2.4）可得：

$$\int_0^a\int_0^b D(x,y)dxdy = \int_0^{\infty} DH(D)dD \tag{9.2.5}$$

如果图像中的物体被阈值灰度级 T 的边界所勾划出来，则物体边界内的 IOD 可由下式给出：

$$IOD(T) = \int_T^{\infty} DH(D)dD \tag{9.2.6}$$

则内部灰度级的平均值为：

$$MGL = \frac{IOD(T)}{A(T)} = \frac{\int_T^{\infty} DH(D)dD}{\int_T^{\infty} H(D)dD} \tag{9.2.7}$$

9.3　点运算

点运算是一种既简单又重要的图像处理技术，它能让用户改变图像上像素点的灰度值，这样通过点运算处理将产生一幅新图像，而且运算后输出图像的每个点的灰度值仅由相应输入像素点的值决定。它不会改变图像内的空间关系。它常用于改变图像的灰度范围及分布。

点运算的特征为：从像素到像素的复制操作。

如：输入图像为 A(x, y)，输出图像为 B(x, y)，则点运算为：

$B(x, y) = f[A(x, y)]$

由上式可以看出，点运算完全由灰度变换函数 f (x) 确定，后者表示了输入灰度级和输出灰度级之间的映射关系。

由于点运算在处理与显示中的重要性，图像处理系统都有专门的处理硬件与之对应，以便能够以视频速率实时完成操作，这样的部件称为查找表（LUT：Look Up Table）。通常图像处理系统都有成组的查找表供编程使用。在通用的计算机上，没有查找表可以直接利用，通过软件逐点处理来实现，这时查找表仅表现为一种数据结构。

9.3.1　点运算的应用

点运算包括亮度增强，对比度拉伸或灰度变换。

在数字化过程中，图像会受到图像传感器的非线性的影响，所获得的灰度值无法正比于光照强度，需使用点运算来调节。如，一幅图像被一个对光照强度成非线性反应的仪器所数字化。点预算可以变换灰度级，使其反映光照强度的等步长增量。

对比度增强：调整亮度、对比度以便观察；当感兴趣的特征仅占据灰度级一个窄的范围，点运算可以扩展感兴趣特征的对比度。有时也称该方法为对比度扩展。

显示标定：突出具有某类特性的像素，平衡传感器的不均匀性。许多显示设备不能保持灰度值和显示亮度之间的线性关系，但在显示前，将点运算和显示非线性组合起来互相抵消，可以保持在显示图像时的线性关系。

轮廓线：点运算可为图像加上轮廓线。可以运用点运算方法进行阈值化，根据灰度级可将一幅图像划分成为一些不连接的区域，为以后的运算确定边界。

裁剪：即将一幅数字图像裁剪为一定灰度级范围的图像，以便存储和显示。因为一幅数字图像通常以整数格式存储，故可用的灰度级范围是有限的，对 8bits 图像，在每个像素值被存储之前，输出灰度级一定要被裁剪到 0-255 的范围之内。运算后，对于灰度级小于 0 的像素点，其灰度级都置为 0，对于灰度级大于 255 的像素点，其灰度级都置为 255。

9.3.2 点运算的种类

1. 线性运算

设 A 为输入图像，B 为输出图像，取灰度变换为：

$$s = f(r) = a \cdot r + b \quad (9.3.1)$$

其中，r 为输入点的灰度值，s 为相应输出点的灰度值，a 称为增益，b 为偏置，如图 9-5 所示。

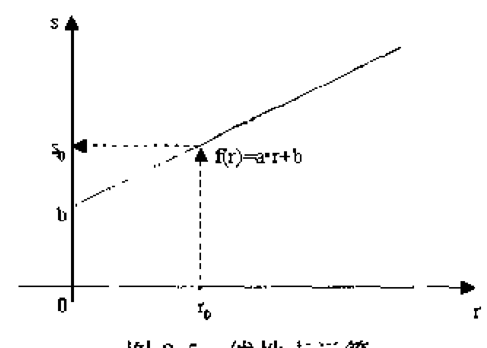


图 9-5 线性点运算

当 a = 1，b = 0 时,将 A(x, y)复制到 B(x, y)。

当 a>1，输出图像的对比度增大。

当 a<1，输出图像的对比度减小。

当 a = 1，b≠0，像素的灰度值上移或下移，整幅图像更亮或更暗。

当 a<0，灰度级反置，暗区域变亮，亮区域变暗。

2. 非线性运算

这类函数保留了函数的基本面貌，但并不限于线性运算，故为非线性点运算。非线性点运算可根据其对中间范围的灰度级的运算而分类。例如，取灰度变换为：

$$f(x) = x + cx(D_m - x) \quad (9.3.2)$$

其中 D_m 为输入图像灰度级的最大值，c （常数）定义了中间灰度增加（c>0）和减少（c0）的程度。那么次灰度变换的特点为：该灰度变换增加中间范围像素的灰度值而只使暗像素值和亮像素值做较小的改变。

令 f(x)如下所示：

$$f(x) = x + \underbrace{cx(D_m - x)}_{\Delta x} \quad (9.3.3)$$

即令Δx = cx(D_m -x)，则有：

当 x 很小时，Δx 小（因为 x 小）。

当 x≈ D_m 时，Δx 也小（因为 D_m -x 小）。

那么在 x 什么的时候，Δx 最大？

由：

$\Delta x' = c(D_m - x) - cx = 0$

得：

$$x = \frac{D_m}{2}$$

又因为：

$\Delta x'' = -2c$

故，当 c>0 时，Δx 在 $x = \frac{D_m}{2}$ 处取得极大值；当 c<0 时，Δx 在 $x = \frac{D_m}{2}$ 处取得极小值。

非线性点运算可以有增加中间范围像素的灰度级而只使暗像素和亮像素做较小改变，用降低较亮和较暗物体的对比度来加强灰度级处于中间范围的物体的对比度，压低在中间灰度级的对比度而在较亮和较暗部分的对比度加强等不同类型。非线性点运算对应于非线性映射函数，典型的映射包括平方函数、对数函数、截取（窗口函数）、阈值函数、多值量化函数等。图 9-6 给出几种典型的非线性点运算的映射函数图。

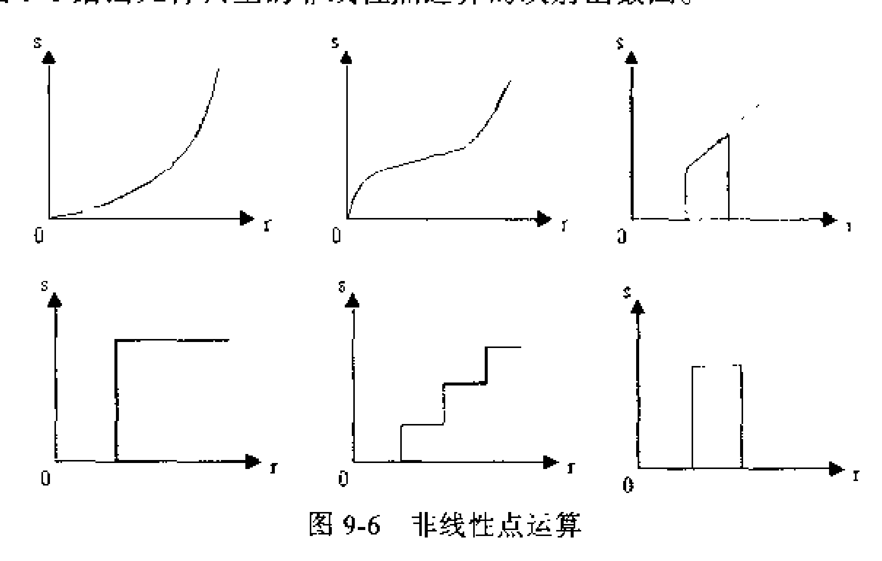


图 9-6　非线性点运算

9.4　点运算和直方图

这一节讨论在已知输入图像直方图和灰度变换函数形式的前提下，预测输出图像直方图的问题。

输出直方图：

给定 A(x, y)，经灰度变换函数 f（d）所定义的点运算后，产生输出 B(x, y)。已知输入图像的直方图 H_A（d），得到输出图像直方图的表达式。即：

A（x y）$\xrightarrow{f(D)}$ B(x, y)

已知 H_A（d）　未知 H_B（d）？

为得到输出图像的直方图，现考虑如下：

任一输出像素的灰度值可由下式得到：

$$D_B = f(D_A) \qquad (9.4.1)$$

如果 f（D）的反函数存在，即：

$$D_A = f^{-1}(D_B) \qquad (9.4.2)$$

（若 f（d）是斜率有限的单调不降的函数，则反函数存在）

图 9-7 给出了输入直方图，灰度变换函数和输出直方图之间的关系。

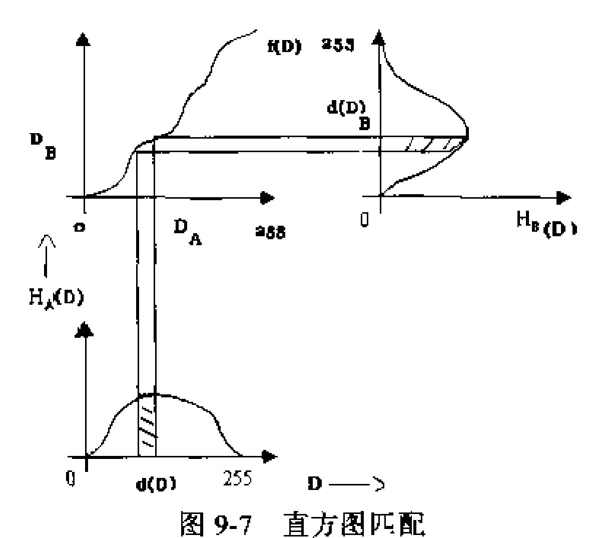


图 9-7 直方图匹配

灰度级由 D_A 向 D_B 转化时，灰度级变换范围也相应转化如下：

$D_A \rightarrow D_B$

$D_A + \Delta D_A \rightarrow D_B + \Delta D_B$

即：D_A 到 $D_A + \Delta D_A$ 之间的所有像素都转换到 D_B 到 $D_B + \Delta D_B$ 中。

有：

$$\int_{D_B}^{D_B+\Delta D_B} H_B(D)dD = \int_{D_A}^{D_A+\Delta D_A} H_A(D)dD \tag{9.4.3}$$

当 ΔD_A 很小时，ΔD_B 也很小。由矩形面积积分近似得：

$H_B(D_B)\Delta D_B = H_A(D_A)\Delta D_A$

则输出直方图：

$$H_B D_B = \frac{H_A(D_A)}{\Delta D_B / \Delta D_A} \tag{9.4.4}$$

$$H_B(D_B) = \lim_{\Delta D_A \to \infty} H_B(D_B) = \lim_{\Delta D_A \to \infty} \frac{H_A(D_A)}{\Delta D_B / \Delta D_A} = \frac{H_A(D_A)}{dD_B / dD_A} = \frac{H_A(D_A)}{(d/dD_A)f(D_A)}$$

即：

$$H_B(D) = \frac{H_A[f^{-1}(D)]}{f'[f^{-1}(D)]} \tag{9.4.5}$$

其中：

$$f' = \frac{df}{dD} \tag{9.4.6}$$

9.5 点运算的应用

9.5.1 直方图均衡化

直方图均衡化经常用于图像比较和分割前的初始化（预处理），主要使用图像对比度

拉伸方法处理。

直方图均衡化是通过灰度变换将一幅图像转换为另一幅具有均衡直方图，即在每个灰度级上都具有相同的像素点数的过程。直方图均衡化处理的中心思想是把原始图像的灰度直方图从比较集中的某个灰度区间变成在全部灰度范围内的均匀分布。

对图像空间域点的增强过程是通过增强函数 t = f（d）来完成的，t、D 分别为目标图像和原始图像上的像素点(x，y)，在进行均衡化处理时，增强函数 f 需要满足两个条件：

（1）增强函数 f（D）在 0⩽D⩽L-1 的范围内是一个单调递增函数，这个条件保证了在增强处理时没有打乱原始图像的灰度排列次序。

（2）需要满足的条件是对于 0⩽D⩽L-1 应当有 0⩽f（D）⩽L-1，它保证了变换过程中灰度值的动态范围的一致性。

同样的，对于反变换过程 $D = f^{-1}(t)$，在 0⩽t⩽1 时也必须满足上述两个条件。

若图像的灰度级的最大值为 D_m，图像面积为 A_0，则直方图均衡化后，每个单位灰度级上像素个数为 A_0/D_m。

则有如下问题：怎样求 f（d），使直方图变化如下？

$H_A = H(d) \rightarrow H_B = A_0/D_m$

根据式（9.4.5）$H_B(D) = \dfrac{H_A[f^{-1}(D)]}{f'[f^{-1}(D)]}$

得：

$$A_0\Big/D_m = \frac{H(D)}{f'(D)}$$

因为：

$$f'(D) = \frac{D_m H(D)}{A_0} \tag{9.5.1}$$

于是：

$$f(D) = \frac{D_m}{A_0}\int_0^D H(u)du \tag{9.5.2}$$

定义 1：图像的概率密度函数（PDF）为被归一化到单位面积的直方图，即：

$$P(d) = \frac{1}{A_0}H(d) \tag{9.5.3}$$

其中，H（d）为直方图，A_0 为图像的面积。

定义 2：图像的累积分布函数（CDF）是其面积归一化的阈值面积函数：

$$P(D) = \int_0^D p(u)du = \frac{1}{A}\int_0^D H(u)du \tag{9.5.4}$$

因此，（9.5.2）式可写为：

$$f(d) = D_m P(d) \tag{9.5.5}$$

即直方图均衡化变换函数(GST)为：

$$D_B = f(D_A) = D_m P(D_A) \tag{9.5.6}$$

或：

$$B(x, y) = f[A(x, y)] = D_m P[A(x, y)]$$

图 9-8 中由上到下分别为：图像，归一化直方图（直方图/面积），累积分布函数。

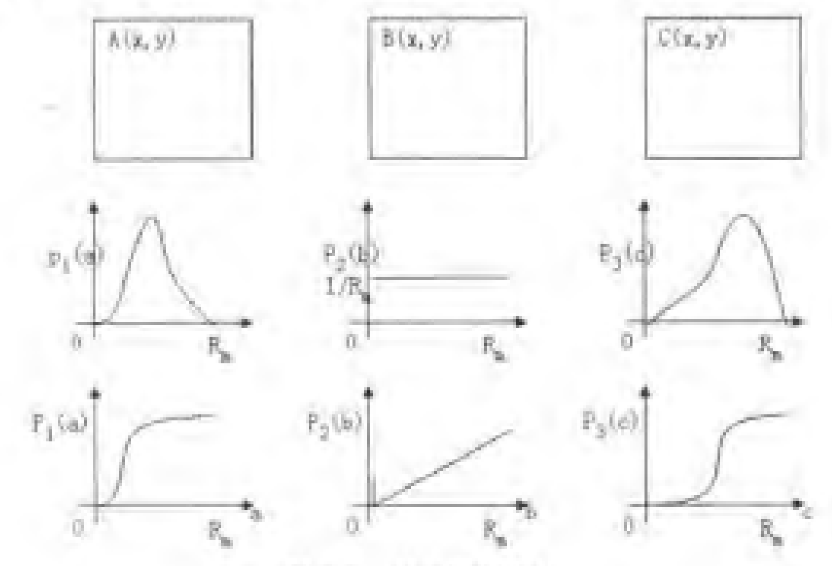


图 9-8 直方图均衡化

从式子（9.5.4），或图 9-8 可以看出，CDF 是个性质好的函数，它是非负的，且处处有非负、有限的斜率，P^{-1} 总存在。

因为灰度级的个数是有限的，故直方图均衡化后，在一些灰度级处可能没有像素，在另外一些灰度级处则像素很拥挤。

图 9-9 是直方图均衡化后的飞机图像及其直方图，可见其直方图与原图的直方图相比是很均衡的，但是，离散情况下不可能做到绝对的一致。

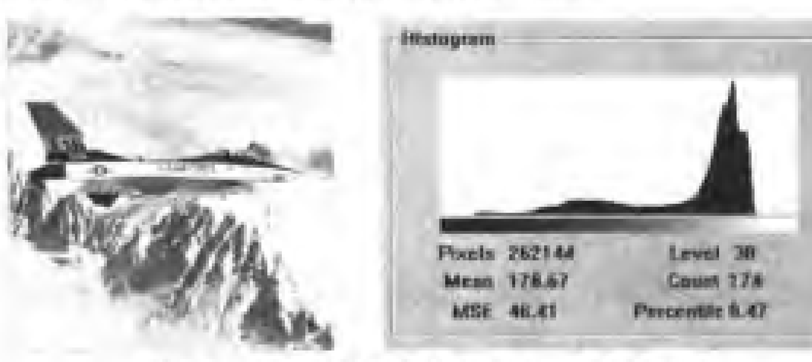


图 9-9 直方图均衡化的飞机图片及其直方图

灰度分布在一个狭窄的区间，细节看不清，如图 9-10 所示。



图 9-10 灰度分布窄的图像及其直方图

直方图均衡化后，灰度间距拉开，增大反差，细节清楚。

9.5.2 直方图匹配

直方图匹配即将某幅影像或某一区域的直方图匹配到另一幅影像上。使两幅影像的色调保持一致。可以在单波段影像直方图之间进行匹配，也可以对多波段影像进行同时匹配。两幅图像比对前，通常要使其直方图形式一致。

例如，同一场景在不同的光照条件下数字化为两幅图像，在对这两幅图像进行比较前，可用直方图变换。

问题：已知输入图像 A(x, y)，要将它变为具有特定直方图（已知）的输出图像 C(x, y) 的形式，求变换 f(d)?

这可以分为两部分完成。第一步，利用点运算 f(D)将 A(x, y)变换为上述的平坦直方图的图像 B(x, y)；第二步，通过第二个函数点运算 g(D)，将 B(x, y)变换为 C(x, y)，即：

$$C(x, y) = g(B(x, y)) \tag{9.5.7}$$

由式（9.5.5），f（d）= D_m P（d），可得为了产生 B(x, y)所需要的条件。综上可得：

$$A(x, y) \rightarrow B(x, y) \Rightarrow B(x,y) = D_m P_1[A(x,y)]$$

$$C(x, y) \rightarrow B(x, y) \Rightarrow B(x,y) = D_m P_3[C(x,y)]$$

$$\Rightarrow D_m P_3[C(x,y)] = D_m P_1[A(x,y)]$$

于是：

$$C(x,y) = P_3^{-1}\{P_1[A(x,y)]\} = g\{f[A(x,y)]\} \tag{9.5.8}$$

即，顺序的先应用 D_m P（d），再应用 g（D），其净效果为零。故有下式成立：

$$g(D) = P_3^{-1}(D/D_m) \tag{9.5.9}$$

小结

本章主要介绍了灰度直方图、直方图的用途、点运算和直方图、点运算的应用等内容。

重点：灰度直方图的定义、意义与应用。

难点：直方图的使用、特定函数形式的图像的直方图的理解和求解及综合性题目的求解。

疑点：如何扩大直方图匹配在图像处理中的应用。

练习九

一、选择题

1. 数字图像的灰度取值范围，即直方图中自变量的取值范围是（　　）。

   A. [0,255]　　B. [-255,0]　　C. [1,255]　　D. [0,254]

2. 如果一个像素点的灰度值为 0，则看到的这个像素点为（　　）色。

   A. 白　　B. 黑　　C. 灰　　D. 红

二、思考题

1. 已知图像大小为 64 × 64，有 8 个灰度级，灰度分布直方图为：

r_k : 0 1/7 2/7 3/7 4/7 5/7 6/7 1

n_k : 790 1023 850 656 329 245 122 81

试计算:

（1）概率密度函数（PDF）.

（2）累积分布函数（CDF）.

（3）直方图均衡化变换函数（GST）.

（4）均衡化后的直方图（PDF）.

2. 假定一幅图像的背景是浅色的，其中有一个深色的物体。如何利用直方图提取物体边界。

三、上机题

1. 在 MATLAB 中，读入一幅图像，并求其灰度直方图.
2. 在 MATLAB 中，读入一幅图像，二值化这幅图像.
3. 在 MATLAB 中，读入一幅图像，进行点运算.

第 10 章　图像分割

前面几章中，介绍了数字图像处理技术的一些背景知识，在本章中，将开始讨论从图像中提取信息的问题，通常称之为图像分析。在图像分析中，输入的是图像，而输出的是对图像或景物的描述。图像分析包括图像分割、特征提取、图像表示、图像描述等，其中，图像分割是图像分析的第一步。

图像分割是指将数字图像分割成互不相交（不重叠）的有意义的子区域的过程。其目的是使各子区域与景物中的各实际物体相对应，以便进行更高层次的图像解释、图像识别等处理。

对灰度图像（本章将主要讨论灰度图像分割）的分割一般是基于图像像素灰度值的基本特性——不连续性和相似性中的一个或两个来进行的。第一类方法是根据图像灰度级的突变将图像进行分区。利用这一类方法，可以对图像中孤立点、线以及图像边缘进行检测；第二类方法是根据事先指定的准则对图像进行分割，得到相似区域。门限处理、区域增长、区域分离和聚合都是这类方法的实例。

本章的主要内容有以下几个方面：

（1）图像的不连续性检测。

（2）边缘连接和边界检测。

（3）阈值处理。

（4）基于区域的分割。

（5）基于形态学分水岭的分割。

（6）运动在图像分割中的作用。

10.1　图像的不连续性的检测

图像有三种基本类型的不连续性，即：点、线、边缘。这一节中，将讨论对这三种不连续性进行检测的方法。寻找不连续性最常用的方法是在对整幅图像使用一个模板进行检测。

在利用模板对图像的不连续性进行检测时，其主要操作是计算出模板系数与它所覆盖的图像区域内对应像素乘积的和。以一个如图 10-1 所示的 3×3 模板为例，在图像中任一点的模板响应可根据下式计算得到：

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

图 10-1　一个一般的 3×3 模板

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9 = \sum_{i=1}^{9} w_i z_i \tag{10.1.1}$$

其中，z 是与模板系数 w 相对应的像素的灰度值。通常，模板的响应是对应于模板的中心位置的。当模板的中心位于图像的边界像素上时，可利用该像素的部分邻域来计算模板的响应。

10.1.1 点检测

对于图像中的孤立点可以很直接地检测到。利用如图 10-1 所示的 3×3 模板。如果：

$$|R|>T \tag{10.1.2}$$

则认为在模板中心的位置上已经检测到一个孤立点，其中，T 是一个非负的阈值，R 根据式（10.1.1）得到。

本质上，该公式计算的是中心像素与其相邻像素的灰度值的加权差。其主要思想是：孤立点的灰度值与其临近像素的灰度值存在很大差别，即：如果一个孤立点（此点的灰度级与其背景的差异相当大并且它所在的位置是一个均匀的或近似均匀的区域）与它周围的点很不相同，则很容易被这类模板检测到。

图 10-2 所示的模板与用作高频空间滤波器的模板在拉普拉斯操作方面是相同的。严格说来，这里强调的是对图像的孤立点的检测。也就是说，只有将灰度差（模板的响应）足够大的那些点视为孤立点才有意义。

-1	-1	-1
-1	8	-1
-1	-1	-1

图 10-2 点检测模板

注意：模板系数之和为零表示在灰度级为常数的区域，模板响应为零。

【例 10-1】图像中孤立点的检测。

以图 10-3（a）所示的 252×252Lena 灰度图为例观察用图 10-2 所示的点检测模板对灰度图像进行孤立点检测的结果。


（a）

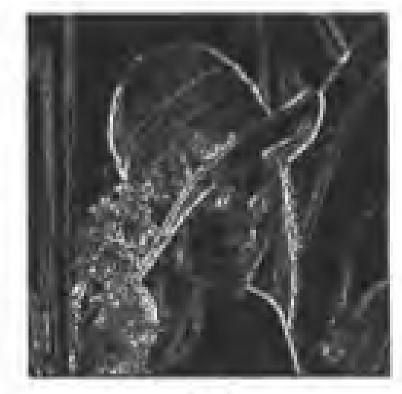

（b）

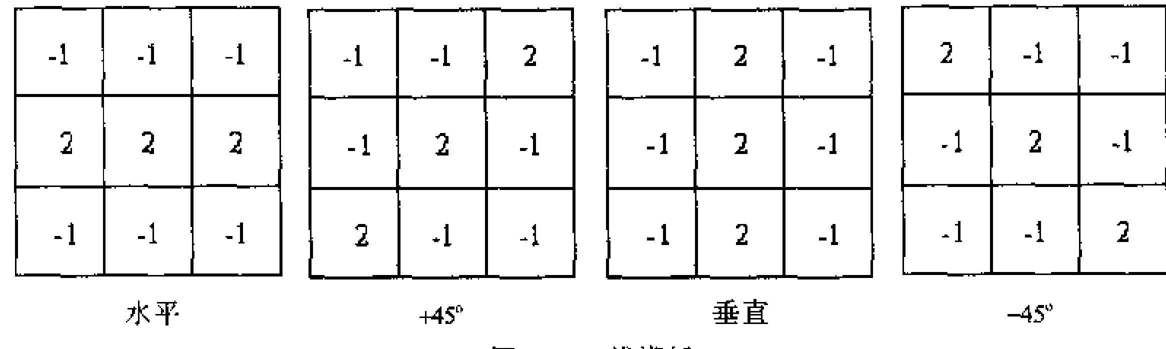
图 10-3 图像孤立点的检测

（a）Lena 图。

（b）用图 10-2 所示模板进行点检测的结果。

10.1.2　线条检测

对图像中线条的检测要比检测图像中的点稍微复杂一些。如图 10-4 中所示模板。



水平

-1	-1	-1
2	2	2
-1	-1	-1

+45°

-1	-1	2
-1	2	-1
2	-1	-1

垂直

-1	2	-1
-1	2	-1
-1	2	-1

-45°

2	-1	-1
-1	2	-1
-1	-1	2

图 10-4　线模板

若将第一个模板在图像上移动，则该模板对水平方向一个像素宽度的线条有更大的响应。

在极限情况下，考虑位于均匀背景上的一条直线，显然，当该线条经过模板的中间一行时会产生最大响应。画出一个除水平行具有不同值外，其他行都为 1 的阵列，就可以很容易验证上述的结论。

通过类似的实验可以发现，图 10-4 中的第二个模板对于 +45°方向的斜线有最佳响应；第三个模板对于垂直方向的直线有最佳响应；第四个模板对于 -45° 方向的斜线有最佳响应。

只要观察每个模板中的主方向（在该方向上模板具有比其他可能方向上更大的系数，此处为 2），同样可以确定模板所检测的线条方向。每个模板系数相加的总和为零，表示在灰度级恒定的区域，模板的响应为零。

令 R1、R2、R3 和 R4 从左到右分别代表图 10-4 中模板的响应，这里 R 的值由式（10.1.1）给出，设四个模板都对整个图像进行了处理。

如果在图像中的某一点，对于所有 $i \neq j$ 均有 $|R_i|>|R_j|$，那么该点较倾向于处在一条方向等于第 i 模板主方向的直线上。

例如，在图像中的某一点，对所有的 $j=2,3,4$，都有 $|R_1|>|R_j|$，那么就可以判定此特定点有可能位于一条水平方向的直线上。

也就是说，如果对检测图像中由给定模板定义的方向上的所有的线感兴趣，只要对整幅图像运用模板，并对得到的结果的绝对值设置门限即可，留下的点就是具有最强响应的点。对于一个像素宽度的线，这些响应最靠近模板定义的对应方向。下述例子说明了这一过程。

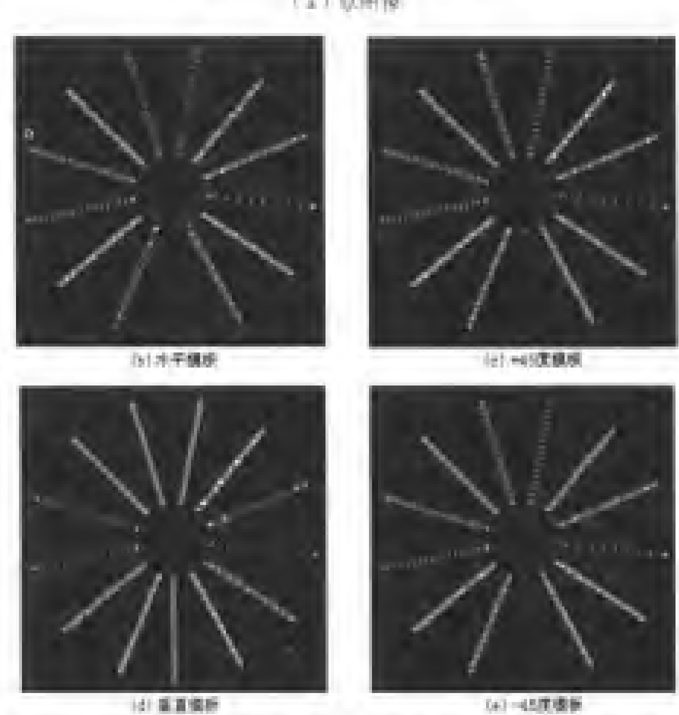
【例 10-2】特定方向上的线检测。

如图 10-5（a）所示是一幅包含各个方向线条的灰度图像。图 10-5（b）~（e）分别为使用如图 10-4 所示的各个线检测模板检测线条的结果。可以明显的看出，在图 10-5（d）中检测得到了垂直方向的线条，而在用其他线模板的检测结果中并未检测出该线条；进一步观察易发现，在各个线检测结果图像中，接近对应模板指定的主方向的线条得到了最强

的响应。



（a）原图像



(b)水平模板　(c)+45度模板

(d)垂直模板　(e)-45度模板

图 10-5　线检测

（a）原图像。

（b）～（e）分别用图 10-4 中的线检测模板的结果。

要决定使用哪一个线拟合模板最好，可以通过设定一个阈值来判断，例如将阈值设为图像中的灰度最大值，便可以容易的判断选择出最佳的线拟合模板，在此不再赘述。

10.1.3　边缘检测

尽管“点”和“线条”的检测是图像分割技术的重要组成部分，但是边缘检测却是检测图像中有意义的不连续性的更常用的方法。这是因为，孤立的“点”和“线条”并不经常存在于图像中。本节将讨论用一阶和二阶数字导数检测一幅图像中边缘的方法。

一条边缘是一组相连的像素集合，这些像素位于两个区域的边界上。

注意：边缘与边界有所不同，一条边缘是一个局部概念，而一个区域的边界是一个更具有整体性的概念。或者可以这样规定边缘，即：边缘需要具有以某种有意义的方式测量灰度级跃变的能力。

从感觉上说，一条理想的边缘应具有如图 10-6（a）所示模型的特性。依据这个模型

生成的完美边缘是一组相连的像素的集合（此处为在垂直方向上），每个像素都处在灰度级跃变的一个垂直的台阶上（如图形中所示的水平线通过图像的灰度剖面图）。

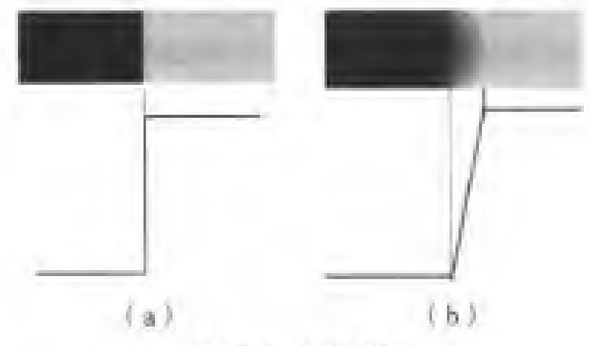


图10-6 边缘模型

（a）理想的数字边缘模型。

（b）斜坡数字边缘模型。

而实际上，由于光学系统、取样和其他图像采集的不完善性，得到的边缘往往是模糊的，模糊的程度取决于诸如图像采集系统的性能、取样率和获得图像的照明条件等因素。结果，边缘被更精确地模拟成具有“类斜面”的剖面，如图10-6（b）所示，在这个模型中，不再有细线（宽为一个像素的线条），而是出现了边缘的点包含斜坡中任意点的情况，并且边缘成为一组彼此相连接的点集。边缘的“宽度”取决于从初始灰度级跃变到最终灰度级的斜坡的长度。这个长度又取决于斜度，斜度又取决于模糊程度。由此可以看到：模糊的边缘使边缘的“宽度”较大，而清晰的边缘使边缘的“宽度”较小。

像素灰度值的不连续性产生边缘，而这种不连续性往往可通过求导数方便地检测到，一般常用一阶导数和二阶导数来检测边缘。如图10-7所示，第一排是一些具有边缘的图像示例，第二排是沿图像水平方向的一个剖面图，第三和第四排分别为剖面的一阶和二阶导数。

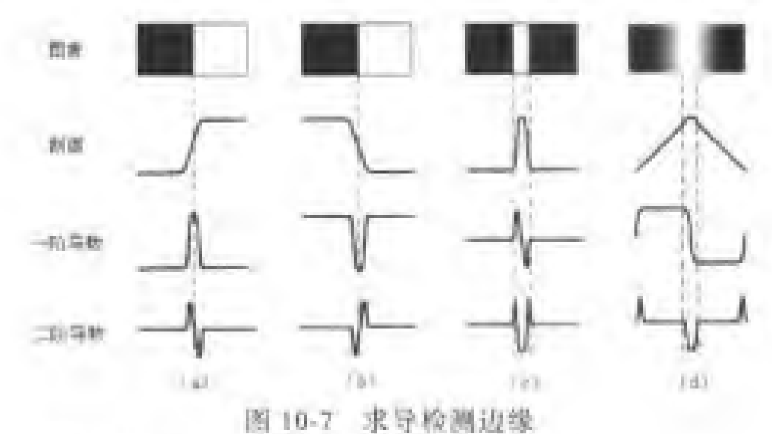


图10-7 求导检测边缘

常见的边缘剖面有三种：

（1）阶梯状（如图10-7（a）和图10-7（b）所示）；

（2）脉冲状（如图10-7（c）所示）；

（3）屋顶状（如图10-7（d）所示）。

阶梯状的边缘处于图像中两个具有不同灰度值的相邻区域之间，脉冲状主要对应细条状的灰度值突变区域，而屋顶状的边缘上升沿和下降沿都比较缓慢。由于采样的缘故，数

字图像中的边缘总有一些模糊，所以这里垂直上下的边缘剖面都表示成有一定坡度。

图 10-7（a）中，对灰度值剖面的一阶导数在图像由暗变明的位置处，有一个向上的阶跃而在其他位置都为零。这表明可用一阶导数的幅度值来检测边缘的存在，幅度峰值一般对应边缘位置。对灰度值剖面的二阶导数在一阶导数的阶跃上升区有一个向上的脉冲，而在一阶导数的阶跃下降区有一个向下的脉冲。在这两个阶跃之间有一个过零点，它的位置正对应原图像中边缘的位置。所以可用二阶导数的过零点检测边缘位置，而用二阶导数在过零点附近的符号确定边缘像素在图像边缘的暗区或明区。分析前页图 10-7（b），可得到相似的结论。这里图像是由明变暗，所以与图 10-7（a）相比，剖面左右对换，一阶导数上下对换，二阶导数左右对换。

前页图 10-7（c）中，脉冲状的剖面边缘与图 10-7（a）的一阶导数形状相同，所以图 10-7（c）的一阶导数形状与图（a）的二阶导数形状相同，而它的两个二阶导数过零点正好分别对应脉冲的上升沿和下降沿。通过检测脉冲剖面的两个二阶导数过零点就可确定脉冲的范围。

前页图 10-7（d）中，屋顶状边缘的剖面可看作是将脉冲边缘底部展开得到的，所以它的一阶导数是将前页图 10-7（c）脉冲剖面的一阶导数的上升沿和下降沿展开得到的，而它的二阶导数是将脉冲剖面三阶导数的上升沿和下降沿拉开得到的。通过检测屋顶状边缘剖面的一阶导数过零点可以确定屋顶位置。

虽然到现在为止，讨论仅限于水平方向上的一维灰度变化，但实际上，对于图像中的任意方向上的边缘都可以进行类似的分析。只要在任意一点简单的定义出垂直于边缘走向的灰度的变化情况，并利用前面的讨论方法就可对结果做出解释。图像中任意点的一阶导数可以利用该点的梯度的幅度来获得，二阶导数可以用拉普拉斯算子得到。

1. 梯度算子

梯度对应一阶导数，梯度算子是一阶导数算子。图像 $f(x,y)$ 在位置 (x,y) 的梯度定义为下列向量：

$$\nabla \vec{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{10.1.3}$$

由向量分析可知，梯度的方向指向点 (x,y) 处 $f(x,y)$ 变化最快的方向。在边缘检测中，这个向量的大小是一个很重要的量，一般的，就简单称其为“梯度”，并用符号 ∇f 表示：

$$\nabla f = mag(\nabla \vec{f}) = \left[G_x^2 + G_y^2 \right]^{1/2} \tag{10.1.4}$$

这个量给出了在 $\nabla \vec{f}$ 方向上每增加单位距离后 $f(x,y)$ 的值增大的最大变化率。另外，梯度向量的方向也是一个重要的量。令 $\alpha(x,y)$ 表示向量 $\nabla \vec{f}$ 在 (x,y) 处的方向角。由向量分析可得：

$$\alpha(x,y) = tg^{-1}\left(\frac{G_y}{G_x} \right) \tag{10.1.5}$$

这里的角度是以 x 轴为基准度量的，在 (x,y) 点处，边缘的方向与该点梯度向量的方向互相垂直。

由方程 10.1.3 和方程 10.1.4 可知，要获得图像的梯度值，必须先计算得到图像每一点的偏导数 $\frac{\partial f}{\partial x}$ 和 $\frac{\partial f}{\partial y}$。令图 10-8（a）中显示的 3×3 大小的区域表示图像邻域中的灰度级，求得 z_5 点处的一阶偏导数的最简单方法之一是使用下列 Roberts 交叉梯度算子（如图 10-8（b）和图 10-8（c）），可以得到：

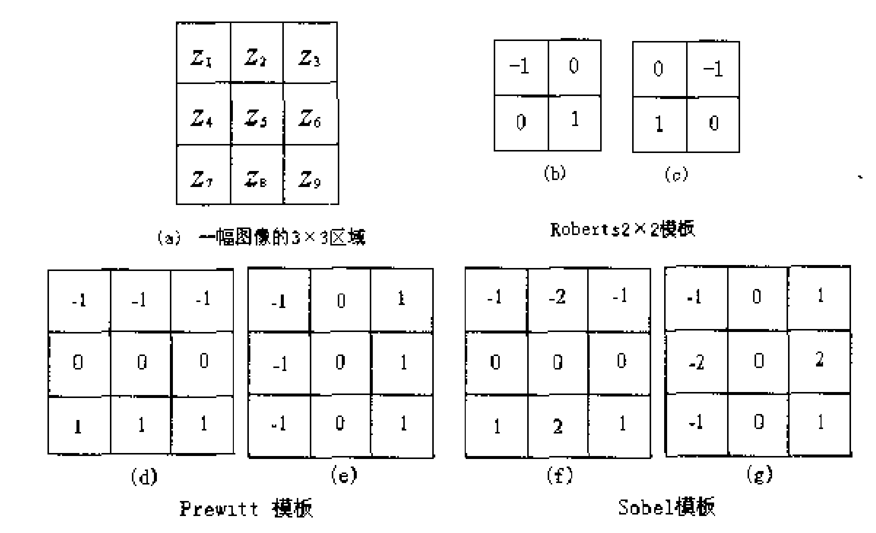


图 10-8　一幅图像的 3×3 区域（z 是灰度值）和用于计算标记为 z_5 的点的梯度的不同模板

$$G_x = z_9 - z_5 \tag{10.1.6}$$

和：

$$G_y = z_8 - z_6 \tag{10.1.7}$$

由于没有鲜明的中心点，2×2 大小的模板很难使用，因此，通常使用 3×3 大小的模板，使用方法由下式给出，图 10-8（d）和（e）中显示的模板（称为 Prewitt 算子）可以用于计算这两个公式：

$$G_x = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$$

和：

$$G_y = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

在这组公式中，3×3 大小的图像区域的第一行和第三行间的差近似于 x 方向上的导数，第三列和第一列之差近似于 y 方向上的导数。

对这两个公式进行一下变化，在中心系数上使用权值 2 可得到：

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \tag{10.1.8}$$

和：

$$G_x = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \tag{10.1.9}$$

权值 2 增加了中心点的重要性，从而实现了某种程度的平滑效果。图 10-8（f）和（g）称为 Sobel 算子，用以实现这两个公式。Prewitt 和 Sobel 算子是在实践中计算梯度时最常用的。Prewitt 模板实现起来比 Sobel 模板更为简单，但后者在噪声抑制特性方面略胜一筹，这在处理导数时是个重要的问题。将模板滑过整幅图像，也即对图像中所有可能的像素位置都应用模板，就可以得到原始图像的梯度图像。当模板作用于图像的边界像素时，用该像素的适当部分的邻近像素进行计算。图 10-8 中，所有模板中的系数总和为 0，表示在灰度级不变的区域，模板响应为 0，正如导数算子所预示的那样。

上面讨论的模板是用于求梯度分量 G_x 和 G_y 的，然而，实际执行中往往因为其计算量太大而使用一种近似计算，即用绝对值对梯度进行近似：

$$\nabla f \approx |G_x| + |G_y| \tag{10.1.10}$$

这个公式在计算方面更方便，而且它仍保持着同灰度级的相对变化。

可以调整图 10-8 中的 3×3 的模板，使它们在沿对角线方向上能有最大的响应。两个用于检测对角线方向上突变的 Prewitt 和 Sobel 算子在下图 10-9 中给出。

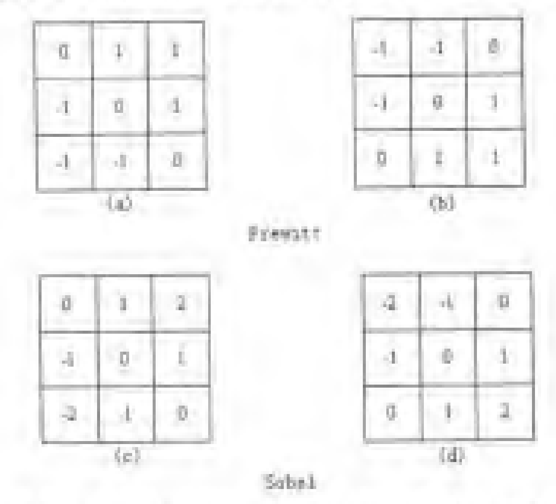
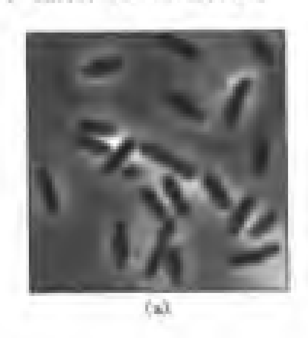
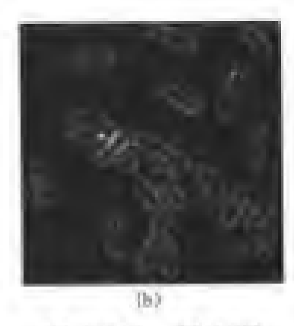


图 10-9 用于检测对角边缘的 Prewitt 和 Sobel 模板

【例 10-3】梯度图示例。

梯度图，如图 10-10 所示。


(a)


(b)

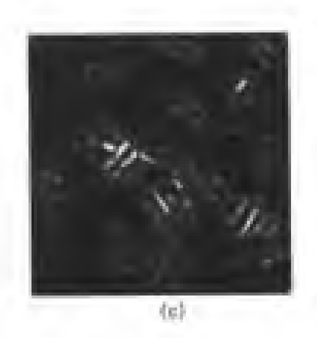

(c)

图 10-10 梯度图

（a）原图。

（b）用图 10-8（d）模板的结果。

（c）用图 10-8（f）模板的结果。

对角线边缘检测，如图 10-11 所示。


(a)


(b)

图 10-11 对角线边缘检测

（a）用图 10-9 中（a）模板的结果。

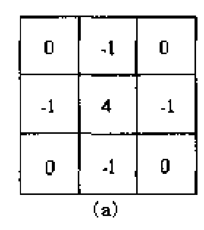
（b）用图 10-9 中（c）模板的结果。

2. 拉普拉斯算子

拉普拉斯（Laplacian）算子是一种二阶导数算子，对一个连续函数 $f(x,y)$，它在 (x,y) 处的拉普拉斯值定义如下：

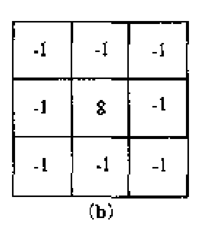
$$\nabla^2 f = \frac{\partial^2 f}{\partial^2 x} + \frac{\partial^2 f}{\partial^2 y} \tag{10.1.11}$$

在数字图像中，也可借助各种模板来计算函数的拉普拉斯值。这里对模板的基本要求是对应中心像素的系数应是正的，而对应中心像素邻近像素的系数应是负的，且它们的和总是零。常用的两种模板分别见图 10-12（a）和（b）。



0	-1	0
-1	4	-1
0	-1	0

(a)



-1	-1	-1
-1	8	-1
-1	-1	-1

(b)

图 10-12　拉普拉斯算子的模板

作为一种二阶导数算子，拉普拉斯算子对图像中的噪声相当敏感，除此之外，它还常产生双像素宽的边缘，而且不能检测边缘的方向。所以，拉普拉斯算子很少直接用于检测边缘。综合以上因素，将拉普拉斯算子在图像分割中的作用总结如下：

（1）利用拉普拉斯的零交叉性质进行边缘定位。

（2）已知边缘像素后，确定该像素是在图像的暗区或明区一边。

对于第一类作用，可将拉普拉斯算子与平滑过程一起利用零交叉作为找到边缘的依据。考虑函数：

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}} \tag{10.1.12}$$

其中，$r^2 = x^2 + y^2$，σ 是标准差。用一幅图像与该函数卷积模糊该图像，图像模糊的程度由 σ 值决定的。h 的拉普拉斯算子，也即 h 关于 r 的二阶导数是：

$$\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}} \tag{10.1.13}$$

因为式（10.1.12）采用了高斯型函数的形式，所以这个公式一般叫做高斯型的拉普拉斯算子（Laplacian of a Gaussian，LoG）。如图 10-13 所示给出了这一函数的二维截面表示。

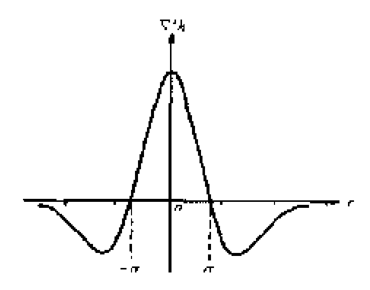


图 10-13　$\nabla^2 h$ 的截面图

容易发现，该函数具有旋转对称性，且具有平滑性质，在 $r = \pm\sigma$ 处取值为零，在中心

部分取正值，在两翼部分取负值。沿相应于竖轴看三维透视图，可以看到，图 10-13 具有经典的墨西哥帽子形状。而图 10-13 正是这种墨西哥帽子的二维剖面图。

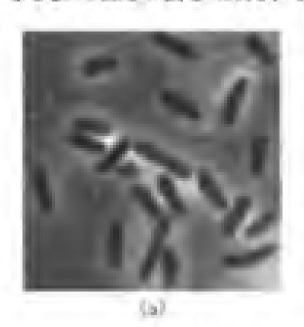
图 10-14 显示了一个对 $\nabla^2 h$ 近似的 5×5 模板，这种近似并不是惟一的。其目的是得到 $\nabla^2 h$ 本质的形状，即：一个正的中心项，周围被一个相邻的负值区域围绕（这个负值区域从原点开始作为距离的函数在值上是增加的），并被一个值为零的外部区域所包围。系数的总和也必须为零，以便在灰度级不变的区域中模板的响应为零，而且这个小的模板仅当图像基本上无噪声时才有用。由于二阶导数是线性运算，所以用 $\nabla^2 h$ 卷积一幅图像与首先使用式（10.1.12）的高斯型平滑函数卷积该图像，然后计算所得结果的拉普拉斯是一样的。因此，看到在 LoG 公式中使用高斯型函数的目的就是对图像进行平滑处理，使用拉普拉斯算子的目的是提供一幅用零交叉确定边缘位置的图像。图像的平滑处理减少了噪声的影响，它的主要作用是抵消由拉普拉斯算子的二阶导数引起的逐渐增加的噪声影响。

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

图 10-14　$\nabla^2 h$ 近似的 5×5 模板

【例 10-4】 通过零交叉寻找边缘。

图 10-15（b）所示的 LoG 结果是为寻找边缘计算零交叉的一幅图像。一种对零交叉点进行近似的直接方法是通过设置 LoG 图像的所有正值区域为白色，负值区域为黑色来阈值化。这种方法的原理在零交叉在拉普拉斯算子的正值和负值之间发生。


(a)


(b)

图 10-15　零交叉检测边缘

（a）原始图像。

（b）LoG。

如果在图像的边缘处，灰度的变化较为剧烈，而且图像的噪声较小，则用高斯拉普拉斯算子对边缘进行检测，可以得到较好的结果。当图像边缘被模糊化了或者当图像中噪声较大时，可以利用拉普拉斯算子的零交叉性质来检测图像边缘的位置。利用拉普拉斯算子的零交叉性质可以对图像的边缘进行精确的定位，而 $\nabla^2 h$ 的平滑特性可以减少噪声的影响。然而，拉普拉斯算子的这些优越性是以增加计算的复杂度和耗费时间为代价的。

10.1.4　组合检测

利用复合模板公式，可以建立一套方法来判定图像中的一个点到底是一个孤立点还是

处于直线上或者边缘上的一个点。为了方便讨论，这里利用矢量的概念。令：

$$\omega = [\omega_1, \omega_2, \ldots, \omega_9]^T \tag{10.1.14}$$

$$z = [z_1, z_2, \ldots, z_9]^T \tag{10.1.15}$$

其中，ω_i 表示一个 3×3 空间模板的系数，z_i 是对应的像素的灰度值，上标 T 表示矢量的转置。利用这一概念，空间模板的响应变为矢量 ω 和矢量 z 的内积。也即：

$$R = \sum_{i=1}^{9} \omega_i z_i = \omega^T z \tag{10.1.16}$$

暂时假设模板具有三个系数，则矢量变成了三维矢量。因此，可以容易地在三维空间看到这些矢量。同时假设用两个模板来检测边缘和线条，并且选择模板的系数使得响应的两个矢量具有单位长度且相互正交。在这种情况下，内积 $\omega_1^T z$ 和 $\omega_2^T z$ 分别等于 z 在矢量 ω_1 和 ω_2 上的投影。其原因在于，对于 ω_1 有：

$$\omega_1^T z = \|\omega_1\| \|z\| \cos\theta \tag{10.1.17}$$

其中，θ 是两个矢量间的夹角，但是，由于 $\|\omega_1\| = 1$，因此有下式：

$$\|z\| \cos\theta = \omega_1^T z \tag{10.1.18}$$

这正是将矢量 z 投影到 ω_1 的结果，如图 10-16 所示。对 ω_2 可作类似讨论。

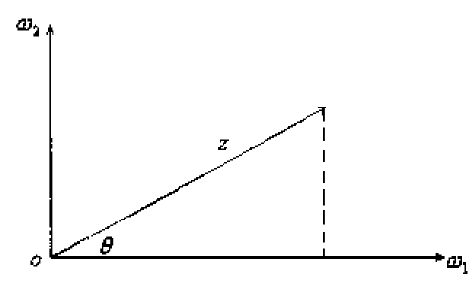


图 10-16　矢量 z 到 ω_1 的投影

假设模为 1 的三个相互正交的矢量：ω_1、ω_2、ω_3 分别相应于 3×3 模板。内积 $\omega_1^T z$、$\omega_2^T z$、$\omega_3^T z$ 分别表示将 z 投影到 ω_1、ω_2 和 ω_3 上。根据上面的讨论，这些内积也代表了三个模板各自的响应。进一步假设模板 1 和 2 代表线条，模板 3 代表边缘。问题是：矢量 z 所代表的区域更像线条还是更像边缘。虽然有两个模板代表线条，但是由于人们只对 z 的线条性质感兴趣，对具体的线条的类型并不感兴趣，因此，只要将 z 投影到由 ω_1 和 ω_2 组成的子空间以及 ω_3 上，就可以解决上述问题。矢量 z 和每一个投影间的夹角标明了 z 到底是和线条还是和边缘子空间接近。图 10-17 的几何关系描述了这一条件。

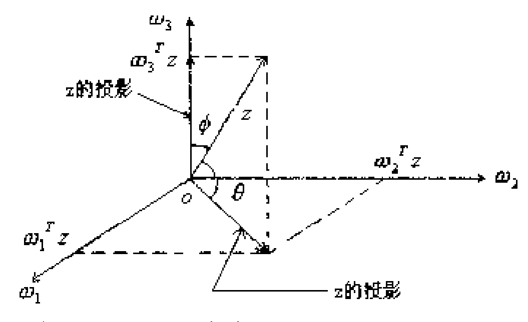


图 10-17　矢量 z 在由 ω_1、ω_2 决定的子空间上的投影以及在 ω_3 上的投影

矢量 z 在由 ω_1 和 ω_2 确定的平面上的幅值为 $\left[(\omega_1^T z)^2 + (\omega_2^T z)^2\right]^{1/2}$。$z$ 的幅值（范数）为：

$$\|z\| = \left[(\omega_1 z)^2 + (\omega_2 z)^2 + (\omega_3 z)^2\right]^{1/2} \qquad (10.1.19)$$

因此，矢量 z 和上述平面（由 ω_1 和 ω_2 确定）间的夹角为：

$$\begin{aligned}\theta &= \cos^{-1}\left\{\frac{\left[(\omega_1{}' z)^2 + (\omega_2{}' z)^2\right]^{1/2}}{\left[(\omega_1{}' z)^2 + (\omega_2{}' z)^2 + (\omega_3{}' z)^2\right]^{1/2}}\right\} \\ &= \cos^{-1}\left\{\frac{\left[\sum_{i=1}^{2}(\omega_i{}' z)^2\right]^{1/2}}{\left[\sum_{i=1}^{3}(\omega_i{}' z)^2\right]^{1/2}}\right\} \\ &= \cos^{-1}\left\{\frac{1}{\|z\|}\left[\sum_{i=1}^{2}(\omega_i{}' z)^2\right]^{1/2}\right\}\end{aligned} \qquad (10.1.20)$$

其中最后一步利用了式（10.1.19）。通过类似的讨论，可以得到 z 与它到 ω_3 子空间的投影间的夹角为：

$$\begin{aligned}\phi &= \cos^{-1}\left\{\frac{1}{\|z\|}\left[\sum_{i=1}^{3}(\omega_i{}' z)^2\right]^{1/2}\right\} \\ &= \cos^{-1}\left\{\frac{1}{\|z\|}\left|\omega_3{}' z\right|\right\}\end{aligned} \qquad (10.1.21)$$

因此，如果 $\theta < \phi$，那么就说由 z 代表的图像区域与线的特征更加接近。

对于一个 3×3 的模板，问题就变成了九维的。上面所讨论的概念仍然适用，但此时需要九个矢量才能组成一个完备的基。图 10-18 中所示的模板满足上面的条件。前四个模板适合于检测边缘，中间四个模板适合于检测线条，最后一个模板（该模板与前面的八个模板组成完备的基）正比于模板所在图像位置的像素的平均值。

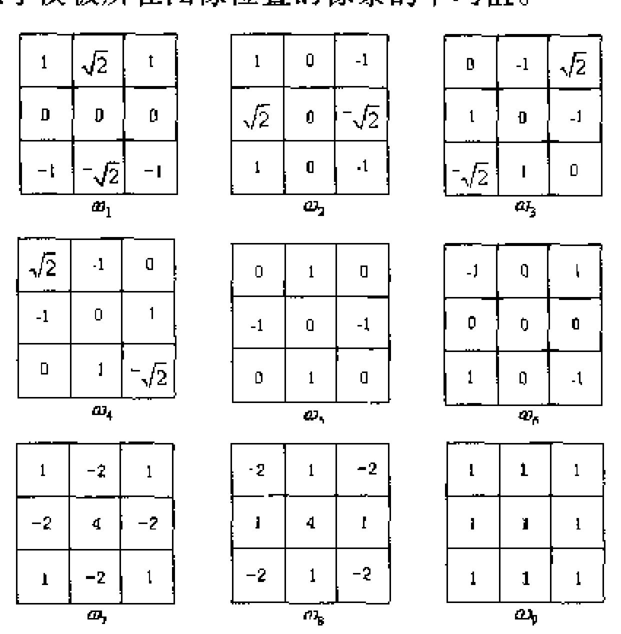


图 10-18 由 $\omega_1, \omega_2, \ldots, \omega_9$ 组成的正交模板系

对于一个用 z 表示的 3×3 的图像区域，可以将矢量 z 在边缘子空间、线条子空间、平

均值子空间的投影的幅值表达成：

$$p_e = \left[\sum_{i=1}^{4}(\omega_i^T z)^2\right]^{1/2} \tag{10.1.22}$$

$$p_l = \left[\sum_{i=1}^{8}(\omega_i^T z)^2\right]^{1/2} \tag{10.1.23}$$

$$p_a = \left|\omega_9^T z\right| \tag{10.1.24}$$

其中 p_e，p_l，p_a 分别表示矢量 z 在上述三个子空间的投影的幅值。这里假设矢量 ω_i（i =1，2，…，9）已经归一化。类似地：

$$\theta_e = \cos^{-1}\left\{\frac{1}{\|z\|}\left[\sum_{i=1}^{4}\left(\omega_i^T z\right)^2\right]^{1/2}\right\} \tag{10.1.25}$$

$$\theta_l = \cos^{-1}\left\{\frac{1}{\|z\|}\left[\sum_{i=5}^{8}\left(\omega_i^T z\right)^2\right]^{1/2}\right\} \tag{10.1.26}$$

$$\theta_a = \cos^{-1}\left\{\frac{1}{\|z\|}\left|\omega_9^T z\right|\right\} \tag{10.1.27}$$

其中 θ_e、θ_l、θ_a 分别表示矢量 z 和它在边缘子空间、线条子空间、平均值子空间投影间的夹角。这些概念也可以直接沿用到其他形式以及其他维数的基，只要这些基之间是正交的。

10.2　边缘连接和边界检测

从理论上讲，前面一节中讨论的方法仅得到在边缘上的像素点。实际上，由于噪声、不均匀的照明而产生的边缘间断以及其他由于引入虚假的亮度间断所带来的影响，往往得到的一组像素不能完整地描绘一条边缘。因此，典型的做法是在使用边缘检测算法后紧跟着进行连接处理将边缘像素组合成有意义的边缘。

10.2.1　局部处理

连接边缘点最简单的方法之一是分析图像中每个点 (x, y) 的一个小邻域（如 3×3 或 5×5）内像素的特点，该点是用前节中讨论过的某一种技术标记了的边缘点。将所有符合事先约定的准则的点（这里认为这些点是相似的点）连接起来，形成由共同符合这些准则的像素组成的一条边缘。

在这种分析过程中确定边缘像素相似性的两个主要性质是：

（1）用于生成边缘像素的梯度算子的响应强度。

（2）梯度向量的方向。

如式（10.1.4）和式（10.1.10）所定义的那样，第一条性质由 ∇f 值给出。因此，如果满足：

$$\left|\nabla f(x,y) - \nabla f(x_0,y_0)\right| \leqslant E \tag{10.2.1}$$

则处在预先定义的 (x, y) 邻域内坐标为 (x_0, y_0) 的边缘像素，在幅度上相似于位于 (x, y) 的像素，这里 E 是一个非负门限。

梯度向量的方向（角度）由式（10.1.5）给出。如果：

$|\alpha(x,y)-\alpha(x_0,y_0)|<A$

则处在预先定义的(x,y)邻域内坐标为(x_0,y_0)的边缘像素具有相似于位于(x,y)的像素的角度。这里A是非负角阈值。如式（10.1.5）所注释的那样，(x,y)处边缘的方向垂直于此点处梯度向量的方向。

如果大小和方向准则得到满足，则在前面定义的(x,y)邻域中的点就与位于(x,y)的像素连接起来。在图像中的每个位置重复这一操作。当邻域的中心从一个像素转移到另一个像素时，这两个相连接点必须记录下来。

10.2.2 运用 Hough 变换进行全局处理

在这一节中，如果点在一条特定形状的曲线上，则先确定再进行连接。与10.2.1节中讨论的局部分析方法不同，现在考虑像素之间的整体关系。

在图像中给出n个点，假设希望找到这些点中位于直线上的点，并将这些点组成子集。一种可行的方法就是先寻找所有由每对点确定的直线，然后找到所有接近特定直线的点组成的子集。这个过程的问题是它涉及寻找$n(n-1)/2\sim n^2$条直线，并且对每个点要与所有直线执行$(n)(n(n-1))/2\sim n^3$次比较，计算量较大，所以不可行，而且没有什么应用价值。

Hough于1962年提出一种替代方法，一般称之为Hough变换，其基本思想是点—线的对偶性（duality）。在图像空间XY中，所有过点(x,y)的直线都满足方程：

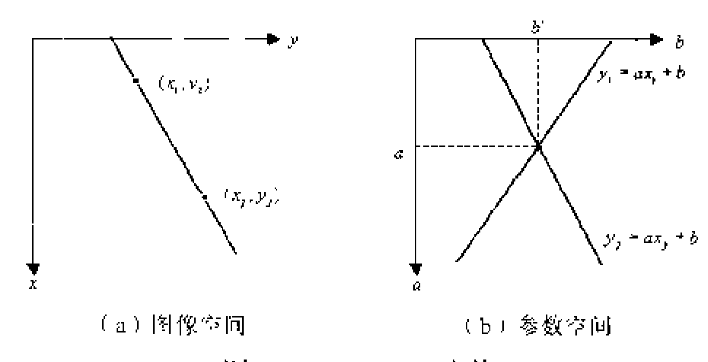
$$y=ax+b \tag{10.2.2}$$

其中a为斜率，b为截距。上式也可写为：

$$b=-xa+y \tag{10.2.3}$$

上式可认为代表参数空间AB中过点(a,b)的一条直线。

如图10-19所示，图（a）为图像空间，图（b）为参数空间。在图像空间XY中过点(x_i,y_i)的通用直线方程按式（10.2.2）可写为$y_i=ax_i+b$，也可按式（10.2.3）写成$b=-x_ia+y_i$，后者表示在参数空间AB里的一条直线。同理过点(x_j,y_j)有$y_j=ax_j+b$，也可写成$b=-x_ja+y_j$，它表示在参数空间AB里的另一条直线。设这两条线在参数空间AB里的点(a',b')相交，这里点(a',b')对应图像空间XY中一条过(x_i,y_i)和(x_j,y_j)的直线，因为它们满足$y_i=a'x_i+b'$和$y_j=a'x_j+b'$。由此可见，图像空间XY中过点(x_i,y_i)和(x_j,y_j)的直线上的每个点都对应在参数空间AB的一些直线，这些直线相交于点(a',b')。



（a）图像空间　　（b）参数空间

图10-19 Hough变换

由此可知，在图像空间中共线的点对应在参数空间里相交的线。反过来，在参数空间

中相交于同一点的所有直线在图像空间里都有共线的点与之对应，这就是点—线的对偶性。Hough 变换就是根据这些关系把在图像空间中的检测问题转换到参数空间里，通过在参数空间里进行简单的累加统计完成检测任务。

在具体计算时需要在参数空间 AB 里建立一个二维的累加数组。设这个累加数组为 $A(a,b)$，如图 10-20 所示，其中$[a_{min},a_{max}]$和$[b_{min},b_{max}]$分别为预期的斜率和截距的取值范围。开始时置数组 A 为零，然后对每一个图像空间中的给定点，让 a 取遍 a 轴上所有可能的值，并根据式（10.2.3）算出对应的 b，再根据 a 和 b 的值（设都已经取整）对 A 累加：$A(a,b)=A(a,b)+1$。累加结束后，根据 $A(a,b)$ 的值就可知道有多少点是共线的，即 $A(a,b)$ 的值就是在 (a,b) 处共线点的个数。同时 (a,b) 值也给出了直线方程的参数，这就得到点所在的直线方程。

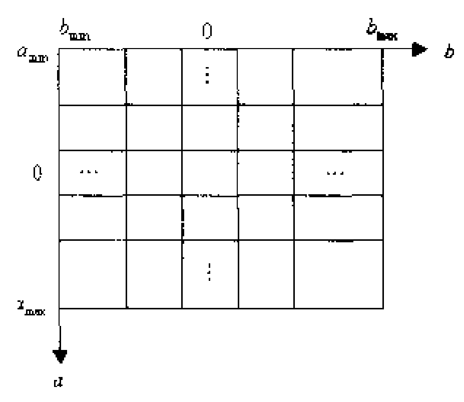


图 10-20　参数空间里的累加数组

（a）直线的一般表示。

（b）将 ρθ 平面量化成累积单元。

注意：这里空间点共线统计的准确性是由累加数组的尺寸决定的。假设把 a 轴分成 K 份（a 取 K 个值），那么对每一个点（x_k,y_k）由式（10.2.3）可得 b 的 K 个值。因为图中有 n 个点，所以这里需要 nK 次运算。可见，运算量是 n 的线性函数。如果 K 比 n 小，则总计算量小于 n 的平方，比前面所述的直接方法要少得多。这就是 Hough 变换的计算优越性所在。运用式（10.2.2）的直线方程时，如果直线接近竖直方向，则会由于斜率和截距的值都接近无穷而使计算量大增（因为累加器尺寸将会很大）。此时，可用直线的极坐标方程（如图 10-21 所示）：

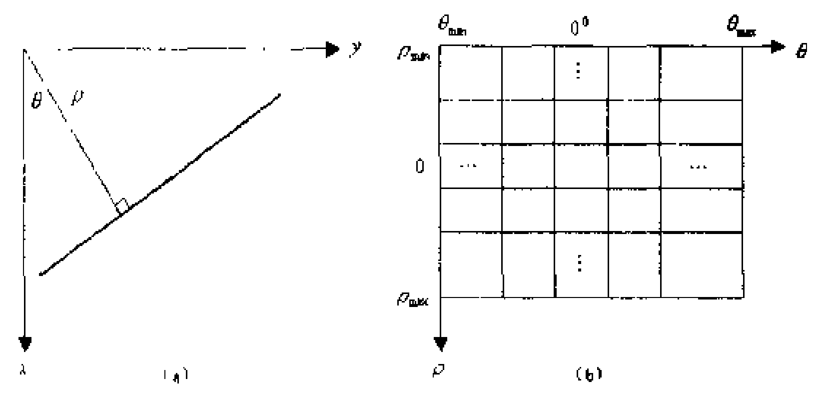


图 10-21　用直线的极坐标方程表示图

$$\rho = x\cos\theta + y\sin\theta \qquad (10.2.4)$$

根据方程（10.2.4），原图像空间中的点对应新参数空间 ΛΘ 中的一条正弦曲线，即原来的点一直线对偶性变成了现在的点一正弦曲线对偶性。检测在图像空间中共点的线需要在参数空间里检测正弦曲线的交点。具体就是让 θ 取遍 Θ 轴上所有可能的值，并根据式（10.2.4）算出所对应的 λ。再根据 θ 和 λ 的值（设都已经取整）对累加数组 A 累加，由 $A(\theta,\lambda)$ 的数值得到共线点的个数。这里在参数空间建立累加数组的方法与上述仍类似，只是无论直线如何变化，θ 和 λ 的取值范围都是有限区间。

尽管到目前为止，人们关注的焦点始终在直线上，但 Hough 变换也适用于任何形式为 $g(v,c)=0$ 的函数，这里 v 是坐标向量，c 是系数向量。

例如，位于圆上的点：

$$(x-c_1)^2+(y-c_2)^2=c_3^2 \qquad (10.2.5)$$

可以通过使用上面讨论的方法进行检测。主要的差别是存在三个参数（c_1，c_2 和 c_3），这三个参数生成了一个带有立方单元和形如 $A(i,j,k)$ 的累加器的三维参数空间。这个过程是增加 c_1 和 c_2，解出满足式（10.2.5）的 c_3，并更新对应于三元组（c_1，c_2 和 c_3）相关单元的累加器。

很明显，Hough 变换的复杂性是与给定函数表达式的坐标和系数的数目成比例的。Hough 变换的进一步推广使人们可以不用简单分析表达式就可以对曲线进行检测，正如灰度级图像的变换的应用。

在边缘连接问题上，一种基于 Hough 变换的连接方法如下：

（1）计算图像的梯度并对其设置门限得到一幅二值图像。

（2）在平面内确定再细分。

（3）对像素高度集中的地方检验其累加器单元的数目。

（4）检验选择的单元中像素之间的关系（主要针对连续性）。

此时，连续性的概念通常基于不连续像素间距离的计算，这些不连续像素是在遍历给定累加器单元对应的像素集合的过程中识别出来的。如果一点和它最接近的相邻点之间的距离超过某一阈值，就认为点之间的间隔是有意义的。

10.2.3 运用图论技术进行全局处理

本节讨论一种全局性的边缘检测方法和基于图形方式表达边缘线段的连接，以及搜索与重要边缘相对应的低开销路径的图形。这种表示提供了一种在有噪声环境下效果很好的抗干扰途径，然而，这个过程更为复杂并且是迄今为止已讨论过的方法中消耗处理时间最大的方法。

先介绍一些基本概念。一个图可表示为 G = {N，A}，其中 N 是一个有限非空的结点集，A 是一个无序结点对的集。集 A 中的每个结点对 (n_i,n_j) 称为一段弧（$n_i\in N,n_j\in N$）。如果图中的弧是有向的，即从一个结点指向另一个结点，则该弧为有向弧，该图称为有向图。当弧是从结点 n_i 指向 n_j 时，那么称 n_j 是子结点，n_i 是 n_j 的父结点。有时父结点也叫祖先，子结点也叫后裔。确定一个结点的各个子结点的过程，称为对该结点的展开或扩展。对每个图还可定义层的概念。第 0 层（最上层）只含一个结点，称为起始结点。最下一层的结点称为目标结点。对任一段弧 (n_i,n_j) 都可定义一个代价（或费用）记为 $c(n_i,n_j)$。如果

有一系列结点 n_1，n_2，…，n_k，其中每个结点 n_i 都是结点 n_{i-1} 的子结点，则这个结点系列称为从 n_1 到 n_k 的一条通路（路径）。

这条通路的总代价为：

$$C=\sum_{i=2}^{k}c(n_{i-1},n_i) \tag{10.2.6}$$

这里定义边缘元素是两个互为4-近邻的像素间的边界，如图10-22（a）中像素 p 和 q 之间的竖线以及图10-22（b）中像素 q 和 r 之间的横线所示。边界是由一系列边缘元素构成的。

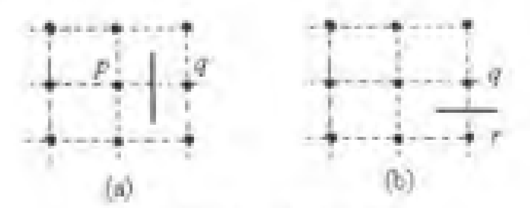


图10-22 边缘元素

现在用图10-23来解释一下如何根据前面所述的概念来检测边界。图10-23（a）给出图像中一个区域，其中圆圈内的数字代表各像素的灰度值。现设每个由像素p和q确定的边缘元素对应一个代价函数：

$$c(p,q)=H-[f(p)-f(q)] \tag{10.2.7}$$

其中H为图像中的最大灰度值（图10-23（a）中为7），$f(p)$ 和 $f(q)$ 为像素 p 和 q 的灰度值。这个代价函数的取值与像素间的灰度值差成反比。按照前面介绍的梯度概念，代价大对应梯度小，代价小对应梯度大。根据式（10.2.7）的代价函数，利用图搜索技术从上向下可检测出如图10-23（b）中所对应大梯度的边界段。

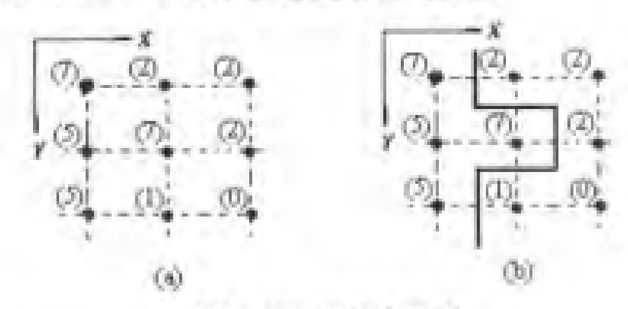


图10-23 图搜索示例

图10-24给出解决这个问题的搜索图。

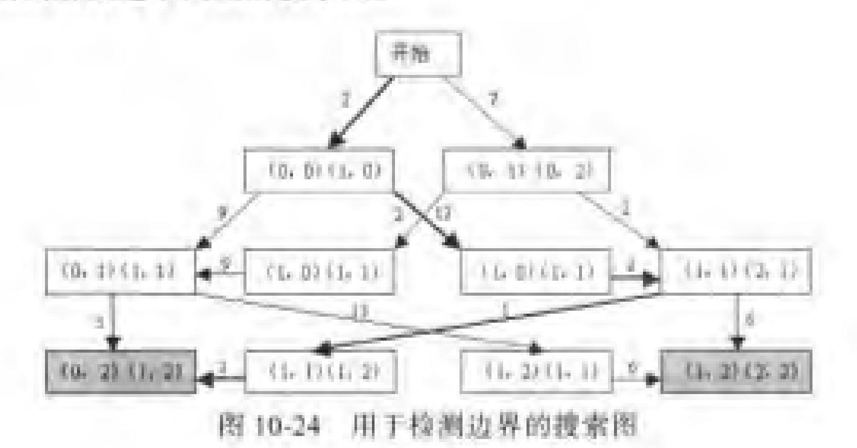


图10-24 用于检测边界的搜索图

每个结点（如前页图 10-24 中用长方框表示）对应一个边缘元素。每个长方框中的两对数分别代表边缘元素两边的像素坐标。有阴影的长方框代表目标结点。如果两个边缘元素是前后连接的，则所对应的前后两个结点之间用箭头连接。每个边缘元素的代价都由式（10.2.7）计算，并标在图中指向该元素的箭头上。这个数值代表了如果用这个边缘元素作为边界的一部分所需要的代价。每条从起始结点到目标结点的通路都是一个可能的边界。图中粗箭头表示根据式（10.2.7）算得的最小代价通路。

如果不利用所需解决问题的一些特征知识，对结点的扩展次序将完全是任意的。由于每个结点都要展开，需扩展的结点数量常常很大，在一般情况下，为求得最小代价所需的计算量通常是很大的，在许多情况下为加快运算速度常常只求次最优。下面介绍一个借助有关具体问题的启发性知识减少搜索的方法。令 $r(n)$ 为从起始结点 s 出发经过结点 n 到达目标结点的最小代价通路的估计代价。这个估计代价可以表示为从起始结点 s 到结点 n 的最小代价通路的估计代价 $g(n)$ 与从结点 n 到目标结点的通路的估计代价 $h(n)$ 之和：

$$r(n)=g(n)+h(n) \tag{10.2.8}$$

这里 $g(n)$ 可取为目前从 s 到 n 的最小代价通路，$h(n)$ 可借助某些启发性知识（例如根据到达某结点的代价确定是否展开该结点）得到。代价的计算可参照前面的方法。根据式（10.2.8）进行图搜索的算法（称为 A^* 算法）由以下几个步骤构成：

（1）将起始结点标记为 OPEN，并置 $g(s)=0$。

（2）如果没有结点被标记为 OPEN，失败退出，否则继续。

（3）将根据式（10.2.8）算得的估计代价 $r(n)$ 为最小的 OPEN 结点标记为 CLOSE。

（4）如果 n 是目标结点，找到通路（可由 n 借助指针上溯至 s）退出，否则继续。

（5）展开结点 n，得到它的所有子结点（如果没有子结点，返回第（2）步）。

（6）如果某个子结点 n_i 还没有标记，置 $r(n_i)=g(n)+c(n,n_i)$，标记它为 OPEN 并将指向它的指针返回到结点 n。

（7）如果子结点 n_i 已标记为 OPEN 或 CLOSE，根据 $g'(n_i)=\min[g[n_i],g(n)+c(n,n_i)]$，更新它的值。将其 g' 值减小的 CLOSE 子结点标记为 OPEN，并将原指向 g' 的所有令其值减小的子结点的指针重指向 n。返回第（2）步。

上述算法通常并不能保证发现全局最小代价通路。它的主要优点是借助启发性知识加快了搜索速度。但已经证明，如果 $h(n)$ 是从结点 n 到目标结点的最小代价通路的代价下界，则上述算法确实可以发现最小代价通路。

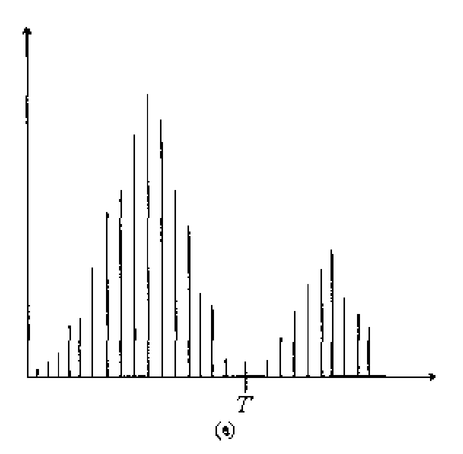
以上算法实际上是将边缘点的检测融进代价函数的计算，从而把边缘检测和边界连接结合起来了。它可用于在给定起点和终点的条件下连接它们之间的边缘段。在图像分割中需检测的区域边界常是闭合的，此时除需确定起始点外，还要解决判断搜索结束的问题。当边界包围的区域比较紧凑（偏心率较小）时，可以通过对图像进行极坐标变换而同时解决确定起始点和判断搜索结束这两个问题，并保证所得边界无卷绕重叠且为单像素宽。这种方法是将动态搜索技术用于状态空间的结果。

10.3 阈值处理

图像阈值处理的直观性和易于实现性使它在图像分割应用中处于中心位置。

10.3.1　基础

假设图 10-25（a）所示的灰度级直方图对应于一幅图像 $f(x,y)$。这副图像由亮的对象和深的背景组成，这样的组成方式将对象和背景具有灰度级的像素分成两组不同的支配模式。



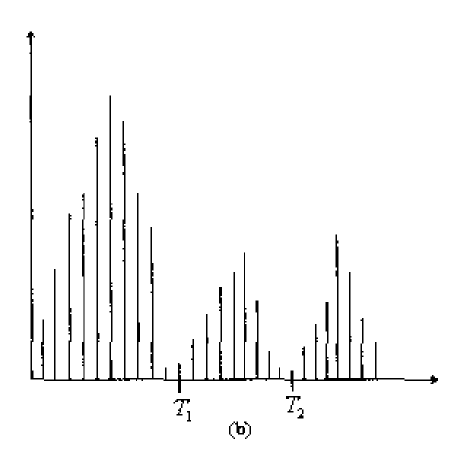


图 10-25　灰度直方图

（a）可以用一个阈值分区的灰度直方图。

（b）可以用两个阈值分区的灰度直方图。

从背景中提取对象的一种显然的方法是选择一个阈值 T，将这些模式分开。然后，所有 $f(x,y)>T$ 的点 (x,y) 称为对象点；否则，就称为背景点。

图 10-25（b）显示了这种方法更为一般化的情况。

这里三个主模式描绘了图像的直方图特性（例如，在暗色背景上的两类亮色对象）。多阈值处理把一个点分类，$T_1<f(x,y)\leqslant T_2$ 时将点分为某一对象，$f(x,y)>T_2$ 时归为另一个对象，$f(x,y)\leqslant T_1$ 时则归为背景。

但是，这种"多级阈值法"并没有对应的单级阈值法那么可靠，因为很难选择一组多级的阈值来有效地对所研究的图像区域进行分离。当与图像对应的直方图具有较多的模式数的时候，这种困难就显得尤为突出。需要多个阈值的分割问题，用 10.4 节将讨论的区域生长方法能得到更好的结果。

基于前面的讨论，取阈值的过程可以看成是对图像的一种操作，这种操作包含了对形如：

$$T=T[x,y,p(x,y),f(x,y)] \tag{10.3.1}$$

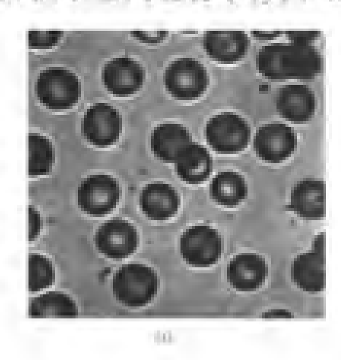
的函数的检验过程。其中 $f(x,y)$ 是点 (x,y) 的像素的灰度值，而 $p(x,y)$ 表示的是该点的某些局域性质，比如点 (x,y) 附近某邻域内的像素的灰度的平均值。一幅经过阈值化处理的图像可以定义为：

$$g(x,y)=\begin{cases}1 & f(x,y)>T\\ 0 & f(x,y)\leqslant T\end{cases} \tag{10.3.2}$$

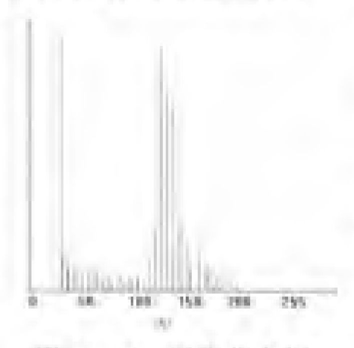
因此，被标记成 1（或者其他方便的强度级别）的像素对应于物体，而被标记成 0 的像素则对应于图像的背景。

当 T 只依赖于图像的灰度值 $f(x,y)$ 时，该阈值称为全局阈值，图 10-26 给出了这种阈值的一个实例。如果 T 既依赖于 $f(x,y)$，又依赖于 $p(x,y)$，该阈值称为局部阈值。另外，

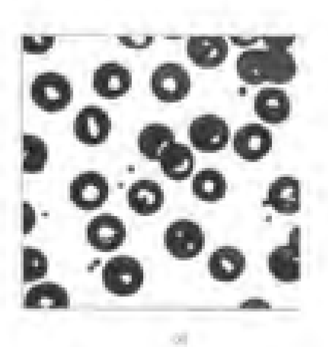
如果 T 依赖于空间坐标 (x, y)，这种阈值称为动态阈值。



(a)



(b)



(c)

图 10-26 阈值化实例

（a）原图。

（b）图像的直方图。

（c）使用全局阈值 T 处理图像得到的结果，此处，阈值取为 100。

10.3.2 简单的全局阈值

全局阈值的处理是阈值处理技术中最简单的：在整个图像中使用单一阈值对图像的直方图进行分割。如果背景的灰度值在整个图像中可合理的看作为恒定，而且所有物体与背景都具有几乎相同的对比度，此时，只要选择了正确的阈值，使用一个固定的全局阈值一般会得到较好的分割效果。

一幅图像，假设其灰度值取值范围在 g_{min} 和 g_{max} 之间，利用阈值方法分割该图像的最简单的步骤如下：

（1）确定灰度阈值 T，且 $g_{min} < T < g_{max}$。

（2）将图像中的像素归类。

将图像中像素的灰度值与阈值 T 相比较，并根据比较结果划为（分割）两类：像素的灰度值大于阈值的为一类，像素的灰度值小于阈值的为另一类，灰度值等于阈值的像素可归入两类中的任意一类。

【例 10-5】 全局阈值处理示例。

前述例子中的阈值是以直方图视觉检测为基础，通过一种试探性的方法确定的。下面的算法可以用于自动地得到 T：

（1）选择一个 T 的初始估计值。

（2）用 T 分割图像。这样做会生成两组像素：G_1 由所有灰度值大于 T 的像素组成，而 G_2 由所有灰度值小于或等于 T 的像素组成。

（3）对区域 G_1 和 G_2 中的所有像素计算平均灰度值 μ_1 和 μ_2。

（4）按下式计算新的阈值。

$$T = \frac{1}{2}(\mu_1 + \mu_2)$$

（5）重复步骤（2）到（4），直到逐次迭代所得的 T 值之差小于事先定义的参数 T。

当背景和对象在图像中占据的面积相近时，好的 T 的初始值就是图像的平均灰度值。如果对象与背景所占的面积不相近时，则其中一个像素组合会在直方图中占主要地位，平均灰度值就不是好的初始选择。此时，T 更合适的初值是诸如灰度值的中间值一类的初值。

但要注意迭代速度和精度的矛盾。

10.3.3 基本自适应阈值

如果成像时的亮度不均匀，会使得本来可以很有效地进行分割的直方图变成用单一全局阈值无法有效分割的直方图。处理这种情况的方法之一就是将图像进一步细分为子图像，并对不同的子图像使用不同的阈值进行分割。这种方法的关键问题是如何将图像进行细分和如何为得到的子图像估计阈值。由于用于每个像素的阈值取决于像素在子图像中的位置，所以说这类阈值处理是自适应的。

10.3.4 最佳全局阈值和自适应阈值

本节将讨论一种产生最小平均分割误差的估计阈值的方法，以及这种方法被应用于解决实际使用阈值处理方法的过程中经常出现的几个重要问题。

假设一幅图像仅包含两类主要的灰度级区域，用 z 表示灰度级值。将这些值看作随机量，则它们的直方图可以被看作它们的概率密度函数（PDF）的估计 $p(z)$。这个总密度函数是两个密度函数的和或者混合，一个是图像中亮区域的密度函数，另一个是暗区域的密度函数，而且，混合参数与相关的亮和暗区域是成比例的。如果密度的表达式已知或进行了假设，则它能够确定一个最佳阈值，将图像分割为两个可区分的区域而使误差最小。

图 10-27 给出了两个概率密度函数。假设两个 PDF 中较大的一个对应于背景的灰度级，而较小的一个描述了图像中对象的灰度级。

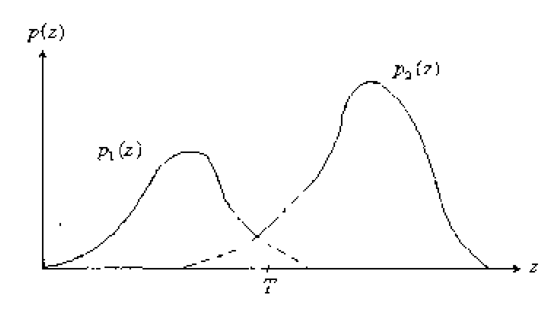


图 10-27 一幅图像中两个区域的灰度级概率密度函数

描述图像中整体灰度级变化的混合概率密度函数是：

$$p(z)=P_1p_1(z)+P_2p_2(z) \qquad (10.3.3)$$

这里 P_1 和 P_2 是两类像素出现的概率，即 P_1 是像素属于对象像素的概率，P_2 是像素属于背景像素的概率。假设任何给定的像素不是属于对象就是属于背景，使得：

$$P_1+P_2=1 \qquad (10.3.4)$$

一幅图像通过将像素灰度值大于门限 T（如上面图 10-27 所示）的像素归类为背景，所有其他像素归类为对象进行分割，关键是选择一个 T 值，使得在决定一个给定的像素是属于对象还是背景时的平均出错率降至最小。

在区间 $[a,b]$ 内取值的随机变量的概率是它的概率密度函数从 a 到 b 的积分，即在这个上下限之间 PDF 曲线围住的面积。因此，将一个背景点当作对象点进行分类时，错误发生的概率为：

$$E_1(T)=\int_{-\infty}^{T}p_2(z)dz \qquad (10.3.5)$$

这是在曲线 $p_2(z)$ 下方位于阈值 T 左边区域的面积。同样，将一个对象点当作背景点进行分类时，错误发生的概率为：

$$E_2(T)=\int_T^{\infty} p_1(z)dz \tag{10.3.6}$$

这是在曲线 $p_1(z)$ 下方位于阈值 T 右边区域的面积。出错率的整体概率是：

$$E(T)=P_2E_1(T)+P_1E_2(T) \tag{10.3.7}$$

注意：公式是如何用对象或背景像素出现的概率对量 E_1 和 E_2 进行加权的，同时应看到它们的脚标是相对应的。公式的这种形式很好解释，例如，考虑背景点从不出现的极端情况，$P_2=0$。因为背景点从不出现，所以将背景点归类为对象点（出错率为 E_1）的概率在整体出错率(E)中的分布可以不加考虑。此时的出错率可以通过用 $P_2=0$ 乘以 E_1 表示。如果背景和对象点以等可能性出现，则权值 $P_1=P_2=0.5$。

要找到出错最少的阈值需要将 $E(T)$ 对 T 求微分（使用莱布尼兹法则）并令微分式等于0。结果是：

$$P_1p_1(T)=P_2p_2(T) \tag{10.3.8}$$

这个等式解出 T，即为最佳阈值。

注意：如果 $P_1=P_2$，则最佳阈值位于曲线 $p_1(z)$ 和 $p_2(z)$ 的交点处（见图 10-27）。

得到一个 T 的分析表达式需要知道两个 PDF 的等式。在实践中，有时并不能对这两个密度进行估计，通常做法是利用参数比较易于得到的密度，这种情况下使用的主要密度之一是高斯密度。高斯密度可以用两个参数完全描述：均值和方差。此时：

$$p(z)=\frac{P_1}{\sqrt{2\pi}\sigma_1}e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}}+\frac{P_2}{\sqrt{2\pi}\sigma_2}e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}} \tag{10.3.9}$$

其中 μ_1 和 σ_1^2 分别是某一类像素（即对象）高斯密度的均值和方差，μ_2 和 σ_2^2 分别是另一类像素的均值和方差。将这个方程用于求解式（10.3.8），得到下列阈值 T 的解：

$$AT^2+BT+C=0 \tag{10.3.10}$$

这里：

$$A=\sigma_1^2-\sigma_2^2$$

$$B=2(\mu_1\sigma_2^2-\mu_2\sigma_1^2) \tag{10.3.11}$$

$$C=\sigma_1^2\mu_2^2-\sigma_2^2\mu_1^2+2\sigma_1^2 2\sigma_2^2\ln(\sigma_2P_1/\sigma_1P_2)$$

由于二次方程有两个可能的解，所以要得到最佳解需要两个阈值。

如果方差都相等，$\sigma^2=\sigma_1^2=\sigma_2^2$，则单一的阈值就足够了：

$$T=\frac{\mu_1+\mu_2}{2}+\frac{\sigma^2}{\mu_1-\mu_2}\ln\left(\frac{P_2}{P_1}\right) \tag{10.3.12}$$

如果 $P_1=P_2$，最佳阈值是均值的平均数。如果 $\sigma=0$，结果也一样。对其他已知形式的密度，比如瑞利和 log-nonnal 密度，决定最佳阈值的方法与上述相似。

不同于为 $p(z)$ 假设一种函数形式，一种最小均方误差方法能由图像的直方图估计一幅图像复合灰度级的 PDF。例如，（连续的）混合密度 $p(z)$ 和（离散的）图像直方图 $h(z_i)$ 之间的均方误差为：

$$e_{ms}=\frac{1}{n}\sum_{i=1}^{n}[p(z_i)-h(z_i)]^2 \tag{10.3.13}$$

这里假设直方图有 n 个点。对全部密度进行估计的主要原因是，要确定 PDF 中是否存在主模式。例如，两个主模式典型地指出在图像（或区域）中存在边缘。这种方式优于计算图像的 PDF。

10.3.5 利用边界特性改进直方图和局部阈值处理

根据前面的讨论，可以很明显地看到：如果直方图的尖峰很高、很窄、具有对称性且被很深的波谷割开，找到“好”的阈值的机会就很大。一种改进直方图形状的方法是只考虑那些位于对象和背景之间边缘上的或在该边缘附近的像素。使用这种方法可以明显看到，直方图对于对象和背景的大小的依赖性变小了。例如，一幅由很小的对象和很大的背景区域组成的图像（或背景很小、对象很大）的直方图中，一个大的尖峰是图像的主要部分，因为某一类像素数目高度集中。而如果只使用位于对象和背景间边缘上的或附近的像素，得到的直方图会有同样高度的尖峰。另外，给定像素位于对象内的概率会与像素位于背景内的概率大体相等，由此改进了直方图尖峰的对称性。最后，正如将在下面段落中说明的那样，使用满足某些简单度量的像素会加深直方图尖峰之间的波谷，这些简单度量是以梯度和拉普拉斯算子为基础的。

上述的这种方法存在的主要问题是：隐含假设对象和背景之间的边缘为已知。很明显，当在对象和背景之间找到的分割线正好全部分割的时候，这一信息在分割中是没有用的。然而，根据 10.1.3 节中的讨论，一个像素是否处在边缘上的迹象可通过计算它的梯度得到。另外，使用拉普拉斯算子还可以知道给定的像素是在边缘亮的一边还是暗的一边。拉普拉斯算子的平均值在边缘的过渡部分处为 0，所以，实际上，根据梯度或拉普拉斯算子准则选择的像素构成的直方图波谷，可以被认为是稀疏分布的。可根据这一特性产生前边讨论过的十分希望的深谷。

在图像中任意点 (x, y) 的梯度 ∇f 由式（10.1.4）或式（10.1.10）给出。同样，拉普拉斯算子 $\nabla^2 f$ 根据前面图 10-12 的模板给出。这两个量可以用于生成一个三级图像，如下所示：

$$s(x,y)=\begin{cases}0 & \nabla f < T \\ + & \nabla f \geqslant T, \quad \nabla^2 f \geqslant 0 \\ - & \nabla f \geqslant T, \quad \nabla^2 f < 0\end{cases} \qquad (10.3.14)$$

这里符号 0、+和 - 表示任意三个不同的灰度级，T 是门限，并且计算每个点 (x, y) 的梯度和拉普拉斯值。对于在亮背景中的暗对象，使用式（10.3.14）生成图像 $s(x, y)$：

其中：

（1）所有不在边缘上的像素（由 ∇f 小于门限 T 确定）标记为 0。

（2）所有在边缘“暗”的一边的像素标记为+。

（3）所有在边缘“亮”的一边的像素标记为 -。

式（10.3.14）中的符号+和 - 对于处在“暗”背景中的“亮”对象是相反的。

从这一过程中得到的信息可以用于生成经过分割的二值图像，图中的“1”对应于有意义的对象，“0”对应于背景。从亮背景到暗对象的过渡（沿着水平或垂直扫描线）必须以 $s(x, y)$ 中一个“ - ”号后跟一个“+”号的形式为特点。对象内部的像素不是标记为“0”就是标记为“+”。最后，从对象反向向背景的过渡用一个“+”号后跟一个“ - ”号的形式表示。

因此，包含对象部分的一条水平或垂直扫描线具有以下结构：

（…）（－，＋）（0或+）（+，－）（…）

这里（…）表示-，+和0的任意组合。最内层的括号包含着对象点，并被标记为1。沿着相同的扫描线除了那些被（－，+）和（+，－）包围着的（0或+）的序列以外，所有其他点都标记为0。

10.3.6 基于不同变量的阈值

迄今为止，人们关注的是阈值处理的灰度值。在某些情况下，传感器产生的用于对图像中每一个像素进行描述的变量不止一个，因此，允许进行多谱段阈值处理。例如：彩色图像，它的每个像素分别由三个RGB值描述。此时，可以构造一个三维"直方图"。基本过程类似于一个变量时使用的方法，例如，对一幅有三个变量的图像（RGB分量），每个像素有16种可能的灰度级，这就构成了一个16×16×16的网格（立方体）。填入每个立方体单元内的数字是RGB分量的值与定义特殊单元位置的坐标相一致的像素编号。每个记入项被图像中像素的总数除，形成一个归一化的直方图。

现在，阈值处理的概念变成在三维空间内寻找点的聚簇的过程。例如，假设在直方图中找到有效点簇K，可以对RGB分量值接近某一个簇的像素赋予一个任意值（如白色的值）而对图像中的其他像素赋予其他的值（如黑色的值），通过这种方式将图像进行分割。这一概念很容易推广至具有更多的分量和更多簇的情况。这种方法的主要困难在于随着变量数目的增加，簇的搜索也变成一种更复杂的工作。可以建立簇的搜索方法，比如：Duda，Hart和Stork［2001］以及Tou和Gonzalez［1974］的著述中都有介绍。

色彩分割是以第14章（彩色图像处理）中介绍的所有颜色模型为基础的。例如，色调和饱和度是很多自动监测应用中很重要的性质。这些性质在试图模拟人所具有的功能时特别重要。比如检测水果是否成熟或检测制造的货物。如第14章（彩色图像处理）中所描述的，色调、饱和度、强度模型（Hue，Saturation，Intensity，HSI）对这类应用是很理想的，因为这种模型非常接近于人类对颜色的感觉方式。使用颜色信号的色调和饱和度分量进行分割的方法具有特别的吸引力，因为它用到的二维数据簇比RGB分割需要的三维簇要易于分析。

10.4 基于区域的分割

分割的目的是将图像划分为不同区域。在10.1节和10.2节中，根据区域间灰度的不连续性通过搜寻区域之间的边界来处理这一问题，而10.3节中分割是通过阈值（以像素性质的分布为基础）来进行的，比如灰度级的值或颜色。在本节中，将讨论以直接找寻区域为基础的分割技术。

10.4.1 基本公式

可以用集合的概念来描述图像分割。令集合R代表原图像整个区域，用集合R_1，R_2，…，R_n表示分割图像得到的n个子区域，且各子区域需满足以下条件：

（1）$\bigcup_{i=1}^{n} R_i = R$。

（2）R_i 是一个连通的区域，$i=1, 2, \cdots, n$。

（3）对所有 i 和 j，$i \neq j$，$R_i \cap R_j = \varnothing$。

（4）$P(R_i)=\text{TRUE}$，对于 $i=1, 2, \cdots, n$。

（5）$P(R_i \cup R_j)=\text{FALSE}$，对于 $i \neq j$。

其中，$P(R_i)$ 是定义在集合 R_i 的点上的逻辑谓词，$\varnothing$ 是空集。

条件（1）说明分割必须是完全的，即每个像素必须属于一个区域；条件（2）要求区域中的点必须与某个预定义的准则相联系；条件（3）说明不同区域必须是不相交的；条件（4）涉及在分割区域内的像素必须满足的性质，例如：当所有 R_i 内的像素有相同的灰度级时，则 $P(R_i)=\text{TRUE}$；条件（5）说明区域 R_i 和 R_j 对于谓词 P 是不同的。

10.4.2 区域生长

区域生长的基本思想是将具有相似性质的像素集合起来构成区域，具体来说，就是先对每个需要分割的区域找一个种子像素作为生长的起点，然后将种子像素周围邻域中与种子像素有相同或相似性质的像素（根据某种事先确定的生长或相似准则来判定）合并到种子像素所在的区域中。将这些新像素当做新的种子像素继续进行上面的过程，直到再没有满足条件的像素可被包括进来。这样一个区域就长成了。

【例 10-6】区域生长的示例。

图 10-28 给出已知种子点进行区域生长的一个示例。图 10-28（a）给出需分割的图像，设已知有 2 个种子像素（标为灰色方块），现要进行区域生长。这里所采用的判断准则是：如果所考虑的像素与种子像素灰度值差的绝对值小于某个门限 T，则将该像素包括进种子像素所在区域。图 10-28（b）给出 T=3 时的区域生长结果，整幅图被较好地分成两个区域；图 10-28（c）给出 T=1 时的区域生长结果，有些像素无法判定；图 10-28（d）给出 T=8 时的区域生长结果，整幅图都被分在一个区域中了。由此可见门限的选择是很重要的。

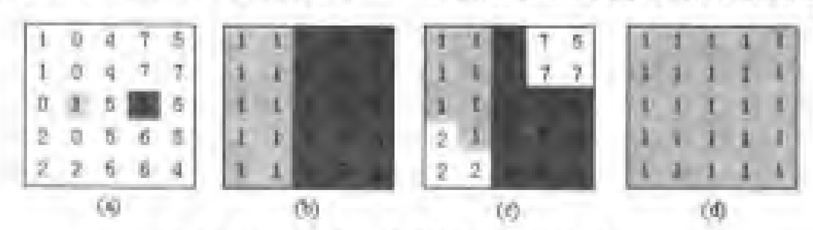


图 10-28　区域生长实例（已知种子点）

在实际应用区域增长法时需要解决三个问题：

（1）根据所解决问题的性质选择一个或多个种子点。

种子像素的选取常可借助具体问题的特点。例如在军用红外图像中检测目标时，由于一般情况下目标辐射较大，所以可选用图中最亮的像素作为种子像素。如果对具体问题没有先验知识，则常可借助生长所用准则对每个像素进行相应计算。如果计算结果呈现聚类的情况则接近聚类中心的像素可取为种子像素。

以图 10-28（a）为例，由它的直方图可知，具有灰度值为 1 和 5 的像素最多且处在聚类的中心，又因为生长准则基于灰度值的差，所以各选一个具有聚类中心灰度值的像素作为种子。

（2）确定在生长过程中能将相邻像素包括进来的准则。

生长准则的选择不仅依赖于具体问题本身，也跟图像数据的类型有关。例如，对地观测卫星成像非常依赖颜色的使用，如果没有彩色图像本身固有的可用信息，对图像的分割会变得非常棘手，甚至无法解决。如果图像是单色的，则必须用一组基于灰度级和空间性质的描绘子（如矩或纹理）对区域进行分析；另外，还需考虑像素间的连通性和邻近性。如果没有将连通性和邻近性信息用于区域生长过程，那么单个的描绘子会产生错误的结果。例如，仅用三个不同的灰度级值将任意一个范围内的像素可视化，具有相同灰度级的像素组成一个“区域”，而不考虑它们的连通性，此时，会生成一个对现在讨论的内容毫无意义的分割结果。

（3）制定让生长停止的条件或规则。

需用公式描述一个终止规则。一般情况下，在没有像素满足生长规则时，区域生长就会停止，但常用的基于像素灰度级、纹理和颜色的生长准则都是基于图像中的局部性质，并没有充分考虑到区域生长的“历史”。为增加区域生长的能力，常需考虑一些与尺寸、形状等图像全局性质有关的准则。

例如：某些准则增强了区域生长算法的处理能力，它们利用了待选像素和已加入生长区的像素间的大小、相似性等概念（比如，待选像素的灰度级和生长区域的平均灰度级之间的比较），以及生长区域的形状。这些类型的描绘子的使用，是以假设能得到预期结果的模型或至少有一部分有效为基础的。

10.4.3 区域分裂与合并

前面介绍的生长方法是从单个种子像素开始，通过不断接纳新像素，最后得到整个区域的。另一种分割方法是从整幅图像开始，通过不断分裂直至最终得到各个区域。实际中通常先把图像分成任意大小且不重叠的区域，然后再合并或分裂这些区域以满足分割的要求。下面介绍一种利用图像四叉树表达方法（如图 10-29 所示）的迭代分裂合并算法。

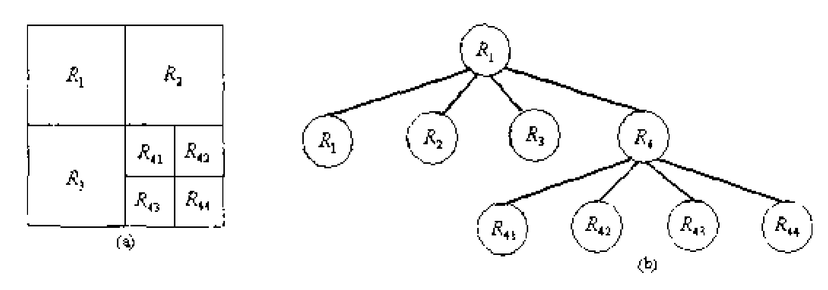


图 10-29 四叉树结构的迭代分裂合并算法

（a）被分割的图像。

（b）对应的四叉树。

根据图 10-29，令 R 代表整个正方形图像区域，P 代表逻辑谓词。对 R 进行分割的一种方法是反复将分割得到的结果图像再次分为四个区域，直到对任何区域 R_i，有 $P(R_i)=\text{TRUE}$。

也就是说，对整幅图像如果 $P(R)=\text{FALSE}$，那么就将图像分成四等份。对任何区域如果有 $P(R_i)=\text{FALSE}$，那么就将 R_i 分成四等份。如此类推，直到 R_i 为单个像素。这种特殊的分割技术用所谓的四叉树形式表示最为方便（即每个非叶子节点正好有四个子树），其中，树的根对应于整幅图像，每个节点对应于划分的子部分。图中，只对 R_4 进行了进一步的再

细分。

如果只是使用分裂，最后有可能出现相邻的两个区域具有相同的性质但并没有合成一体的情况。为解决这个问题，允许拆分的同时进行区域合并，即在每次分裂后允许其继续分裂或合并。这里的“合并”是指合并那些相邻且合并后组成的新区域满足逻辑谓词 P 的区域。换句话说，如果能满足 $P(R_i \cup R_j)=\text{TRUE}$，则将 R_i 和 R_j 合并起来。

上述讨论可以总结为如下过程：在反复操作的每一步，需要：

（1）对于任何区域 R_i，如果 $P(R_i)=\text{FALSE}$，就将每个区域都拆分成四个相连的象限区域。

（2）将 $P(R_i \cup R_j)=\text{TRUE}$ 的任意两个相邻区域 R_i 和 R_j 进行合并。

（3）当再无法进行聚合或拆分时操作停止。

可以对前面讲述的基本思想进行几种变化。例如，一种可能的变化是开始时将图像拆分为一组图像块，然后对每个图像块进一步进行上述拆分，但对开始时的聚合操作有所限制，只能将四个块并为一组，这四个块是四叉树表示法中节点的后代且都满足逻辑谓词 P。当不能再进行此类合并时，这个过程终止于满足步骤2的最后的区域合并。在这种情况下，合并的区域可能会大小不同。这种方法的主要优点是对于分裂和合并都使用同样的四叉树，直到合并的最后一步。

【例10-7】分裂合并法示例。

图10-30表示了使用分裂合并法分割图像的步骤。设图中阴影区域为目标，白色区域为背景，它们都具有常数灰度值。对整个图像 R，$P(R)=\text{FALSE}$（这里令 $P(R)=\text{TRUE}$ 代表在 R 中的所有像素都具有相同的灰度值），所以先将其分裂成如图10-30（a）所示的四个正方形区域。由于左上角区域满足 P，所以不必继续分裂。其他三个区域继续分裂而得到图10-30（b）。此时除包括目标下部的两个子区域外其他区域都可按照目标和背景分别进行合并。对那两个子区域继续分裂可得到图10-30（c）。因为此时所有区域都已满足 P，所以最后一次合并就可得到如图10-30（d）所示的分割结果。

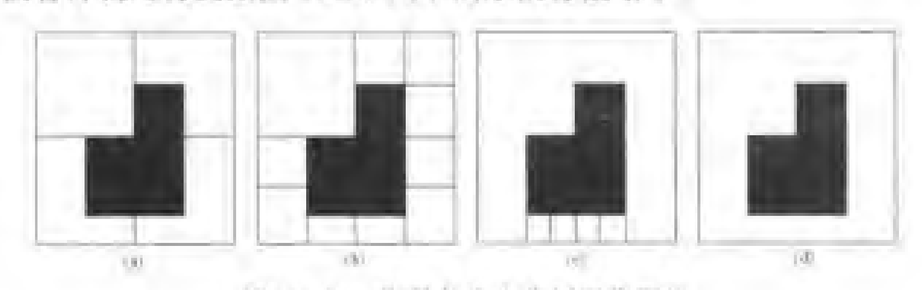


图10-30　分裂合并法分割图像图解

10.5　基于形态学分水岭的分割

到现在为止，已经讨论了基于三种主要概念的分割方法：间断的检测、阈值处理、区域处理。每种方法各有优缺点，例如，全局阈值处理的速度较快；以灰度级的间断检测为基础的方法需要诸如边线连接等后续处理等等。本节中，将讨论基于形态学分水岭概念的方法。形态学分水岭分割将其他三种方法中的许多概念进行了具体化，包括连续的边界分割在内，它生成的分割结果通常更为稳定。这条途径为在分割过程中结合基于知识的约束提供了一个简单的框架。

10.5.1 基本概念

分水岭的概念是以对图像进行三维可视化处理为基础的，其中两维是坐标，另一维是灰度级。对于这样一种“地形学”的解释，这里考虑三类点：

（1）属于局部性最小值的点。

（2）当一滴水放在某点上的时候，水一定会下落到一个单一的最小值点。

（3）当水处在某个点上时，水会等概率地流向不止一个这样的最小值点。

对一个特定的区域最小值，满足条件（2）的点的集合称为这个最小值的“汇水盆地”或“分水岭”。满足条件（3）的点的集合组成地形表面的峰线，术语称做“分割线”或“分水线”。

这些思想可以用图 10-31 作为辅助，做进一步的解释。

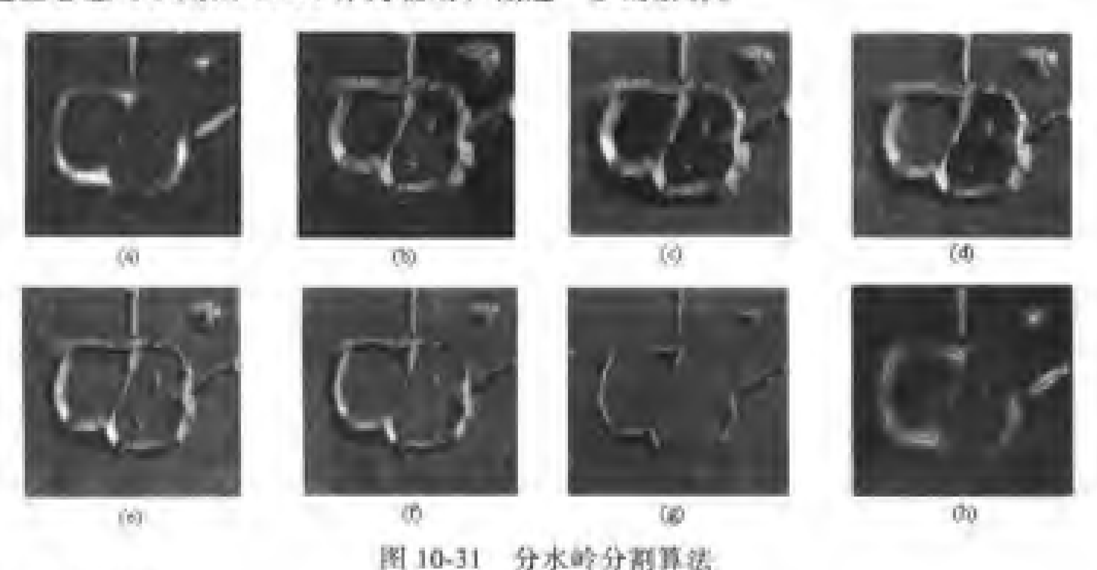


图 10-31 分水岭分割算法

（a）原图像。

（b）地形俯视图。

（c）～（d）被水淹没的两个不同阶段。

（e）进一步淹没的结果。

（f）汇水盆地的水开始聚合（它们之间有一条短坝）。

（g）长一些的水坝。

（h）最后的分水线（分割）。

图 10-31（a）显示了一个简单的灰度图像。图 10-31（b）是地形图。其中“山峰”的高度与输入图像的灰度值成比例。为了易于解释，这个结构的后方被遮蔽起来，以免与灰度级值相混淆。三维表达对一般地形学非常重要，为了阻止上升的水从这些结构的边缘溢出，可以想象将整幅地形图的周围用比最高山峰还高的大坝包围起来。最高山峰的值由输入图像灰度值可能具有的最大值决定。

假设在每个区域最小值中打一个洞（如图 10-31（b）中的深色区域），并且让水以均匀的上升速率从洞中涌出，从低到高淹没整个地形。图 10-31（c）说明被水淹没的第一个阶段，这里水用浅灰色表示，覆盖了对应于图中深色背景的区域。在图 10-31（d）和图 10-31（e）中，可以看到水分别在第一和第二汇水盆地中上升。由于水持续上升，最终水将从一

个汇水盆地中溢出到另一个汇水盆地之中。图 10-31（f）中显示了溢出的第一个征兆。这里，水确实从左边的盆地溢出到右边的盆地，并且两者之间有一个短“坝”（由单像素构成）阻止这一水位的水聚合在一起（在接下来的章节中将讨论坝的构筑）。由于水位不断上升，实际的效果要超出所说的，如图 10-31（g）所显示的那样，两个汇水盆地之间显示了一条更长的坝，另一条水坝在右上角。这条水坝阻止了盆地中的水和对应于背景的水的聚合。这个过程不断延续直到到达水位的最大值（对应于图像中灰度级的最大值）。水坝最后剩下的部分对应于分水线，这条线就是要得到的分割结果。

对于这个例子，在图 10-31（h）中显示为叠加到原图上的一个像素宽的深色路径。分水岭分割算法的一个很重要的性质就是分水线组成一条连通的路径，由此给出了区域之间的连续的边界。

分水岭分割算法的主要应用是从背景中提取近乎一致（类似水滴的）的对象。那些在灰度级上变化较小的区域的梯度值也较小。因此，实际上，经常可以见到分水岭分割方法与图像的梯度有更大的关系，而不是图像本身。有了这种表示方法，汇水盆地的局部最小值就可与对应于所关注对象的小的梯度值联系起来了。

10.5.2 水坝构造

首先看一下如何构造分水岭分割法所需的水坝或者分水线。水坝的构造是以二值图像为基础的，这种图像属于二维整数空间 Z^2。构造水坝分离二元点集的最简单的方法是使用形态膨胀。

如图 10-32 所示说明了如何使用形态膨胀来构造水坝。图 10-32（a）显示了两个汇水盆地的部分区域在淹没步骤的第 n-1 步时的图像。图 10-32（b）显示了淹没的下一步（第 n 步）的结果。水已经从一个盆地溢出到另一个盆地，所以，必须建造水坝阻止这种情况的发生。为了与后面符号相一致，令 M_1 和 M_2 表示在两个区域极小值中包含的点的坐标集合。然后，将处于汇水盆地中的点的坐标集合与这两个在溢出的第 n-1 个阶段的最小值联系起来，并用 $C_{n-1}(M_1)$ 和 $C_{n-1}(M_2)$ 表示。这就是图 10-32（a）中的两个黑色区域。

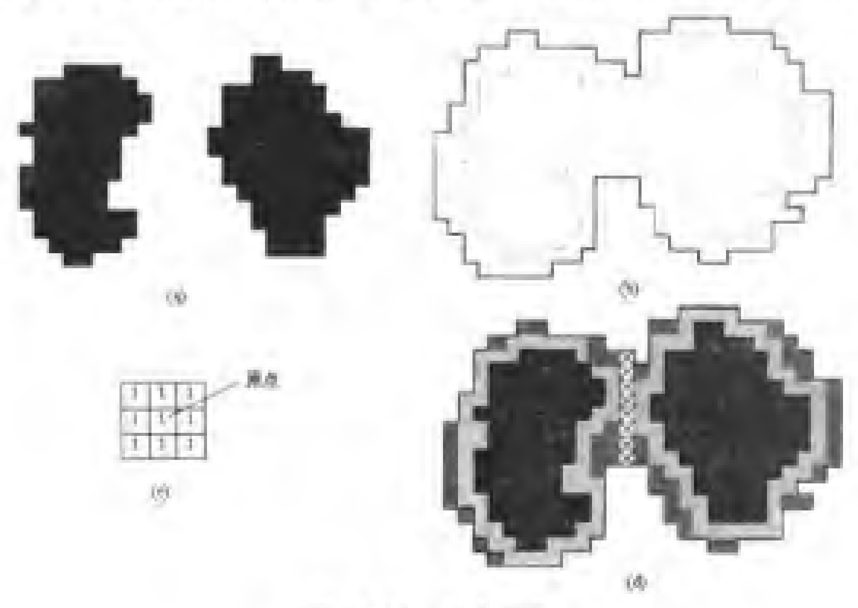


图 10-32 形态膨胀

（a）在淹没的第 n-1 个阶段淹没的汇水盆地的两个部分。

（b）淹没的第 n 阶段，显示出两个盆地间的水已经溢出（水用白色表示）。

（c）用于膨胀的结构元素。

（d）扩展的结果和水坝的构造。

令这两个集合的联合用 $C[n-1]$ 表示。图 10-32（a）中有两个连通分量，而图 10-32（b）中只有一个连通分量。这个连通分量包含着前面的两个分量，用虚线表示。两个连通分量变成一个连通分量的事实说明两个汇水盆地中的水在淹没的第 n 步聚合了。用 q 表示此时的连通分量。

注意：第 n-1 步中的两个连通分量可以通过用"与"操作（$q \cap C[n-1]$）从 q 中提取出来。人们也注意到，属于独立的汇水盆地的所有点构成了一个单一的连通分量。

假设图 10-31（a）中的每个连通分量利用图 10-32（c）中显示的结构元膨胀，需满足两个条件：

（1）膨胀受到 q 的约束（这意味着在膨胀的过程中结构化元素的中心只能定位于 q 中）。

（2）在引起集合聚合的那些点上不能执行膨胀（以防成为单一的连通分量）。

图 10-32（d）使用每个初始连通分量的边界，显示了首轮膨胀（用浅灰色表示）。

注意：在膨胀过程中每个点都满足条件（1），因此，每个区域的边界都进行了均匀的扩展；在膨胀处理过程中，条件（2）并没有应用于任何点。

在第二轮膨胀中（中等灰度表示），几个不满足条件（1）的点符合条件（2）时，得到图中显示的断开周界。很明显，只有满足上述两个条件的属于 q 中的点描绘了图 10-32（d）中交叉阴影线表示的一个像素宽度的连通路径。这条路径组成在淹没的第 n 个阶段希望得到的水坝。在这个淹没水平上，所有刚好在这条路径上的点的值为比图像中灰度值的最大值还大的值。这样，就完成了水坝的构造。所有水坝的高度通常设定为图像中灰度值最大允许值再加上 1。这样设定可以阻止在水位不断升高的情况下，水越过部分水坝。应该特别注意的是，通过这一过程建立的水坝是连通分量，就是希望得到的分割边界，也即：这种方法消除了分割线产生间断的问题。

对于更为复杂情况，处理方法是完全相同的，包括图 10-32（c）中显示的 3×3 对称结构元素的使用也是相同的。

10.5.3 分水岭分割算法

令 M_1，M_2,…、M_R 为表示图像 $g(x,y)$ 的局部最小值点的坐标的集合。如同在 10.5.1 节结尾说明的那样，这是一幅典型的梯度图像。令 $C(M_i)$ 为一个点的坐标的集合，这些点位于与局部最小值 M_i（无论哪一个汇水盆地内的点都组成一个连通分量）相联系的汇水盆地内。符号 min 和 max 代表 $g(x,y)$ 的最小值和最大值。最后，令 $T[n]$ 表示坐标 (s,t) 的集合，其中 $g(s,t)<n$ 即：

$$T[n]=\{(s,t)\mid g(s,t)<n\} \qquad (10.5.1)$$

在几何上，$T[n]$ 是 $g(x,y)$ 中的点的坐标集合，集合中的点均位于平面 $g(x,y)=n$ 的下方。

随着水位以整数量从 n = min+1 到 n = max+1 不断增加，图像中的地形会被水漫过。在水位漫过地形的过程中的每一阶段，算法都需要知道处在水位之下的点的数目。从概念上

来说，假设$T[n]$中的坐标处在$g(x,y)=n$的平面之下，并被“标记”为黑色，所有其他的坐标被标记为白色，然后，当人们在水位以任意增量n增加的时候，从上向下观察xy平面，会看到一幅二值图像。在图像中黑色点对应于函数中低于平面$g(x,y)=n$的点。这种解释对于理解下面的讨论很有帮助。

令$C_n(M_i)$表示汇水盆地中点的坐标的集合。这个盆地与在第n阶段被淹没的最小值有关。参考前一段的讨论，$C_n(M_i)$也可以被看作由下式给出的二值图像：

$$C_n(M_i)=C(M_i)\cap T[n] \tag{10.5.2}$$

也就是说，如果$(x,y)\in C(M_i)$且$(x,y)\in T[n]$，则在位置(x,y)有$C_n(M_i)=1$。否则$C_n(M_i)=0$。对于这个结果几何上的解释是很简单的，人们只需在水溢出的第n个阶段使用“与（AND）”算子将$T[n]$中的二值图像分离出来即可。$T[n]$是与局部最小值M_i相联系的集合。

接下来，令$C[n]$表示在第n个阶段汇水盆地被水淹没的部分的并集：

$$C[n]=\bigcup_{i=1}^{R}C_n(M_i) \tag{10.5.3}$$

然后令$C[\max+1]$为所有汇水盆地的并集：

$$C[\max+1]=\bigcup_{i=1}^{R}C(M_i) \tag{10.5.4}$$

可以看出，处于C_nM_i和$T[n]$中的元素在算法执行期间是不会被替换的，而且这两个集合中的元素的数目与n保持同步增长。因此，$C[n-1]$是集合$C[n]$的子集。根据式（10.5.2）和式（10.5.3），$C[n]$是$T[n]$的子集，所以，$C[n-1]$是$T[n]$的子集。从这个结论可得出重要的结果：$C[n-1]$中的每个连通分量都恰好是$T[n]$的一个连通分量。

找寻分水线的算法开始时设定$C[\min+1]=T[\min+1]$。然后算法进入递归调用，假设在第n步时，已经构造了$C[n-1]$。根据$C[n-1]$求得$C[n]$的过程如下：令 Q 代表$T[n]$中连通分量的集合。然后，对于每个连通分量$q\in Q[n]$，有下列三种可能性：

（1）$q\cap C[n-1]$为空。

（2）$q\cap C[n-1]$包含$C[n-1]$中的一个连通分量。

（3）$q\cap C[n-1]$包含$C[n-1]$多于一个的连通分量。

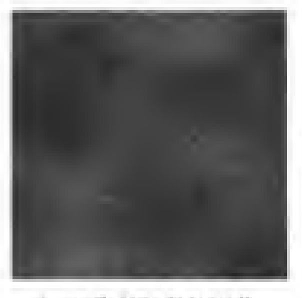
根据$C[n-1]$构造$C[n]$取决于以上三个条件。当遇到一个新的最小值符合条件（1）时，则将q并入$C[n-1]$构成$C[n]$；当q位于某些局部最小值构成的汇水盆地中时，符合条件（2），此时将q合并入$C[n-1]$构成$C[n]$；当遇到全部或部分分离两个或更多汇水盆地的山脊线的时候，符合条件（3）。进一步的注水，会导致不同盆地的水聚合在一起，从而使水位趋于一致，因此，必须在q内建立一座水坝（如果涉及多个盆地就要建立多座水坝）以阻止盆地内的水溢出。正如前一节中的解释，当用3×3个1的结构元素膨胀$q\cap C[n-1]$并且将这种膨胀限制在q内时，一条一个像素宽度的水坝就能够被构造出来。

通过使用与$g(x,y)$中存在的灰度级值相对应的n值，可以改善算法效率；根据$g(x,y)$的直方图，可以确定这些值及其最小值和最大值。

【例 10-8】分水岭分割算法示例。

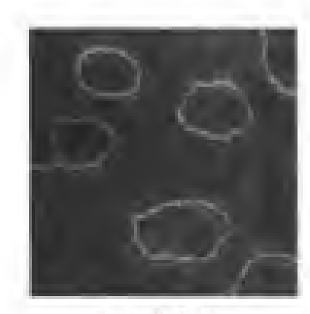
分别考虑图 10-33（a）和（b）中显示的图像和它的梯度。应用上面讨论的分水岭算法得到图 10-33（c）中显示的梯度图像的分水线（白色路径）。将分割的边界叠加在原图

上，如图 10-33（d）所示。正如本节开始时提到，分割边界具有那些被连接起来的路径的重要性质。



（a）带有斑点的图像


（b）梯度图像


（c）分水线

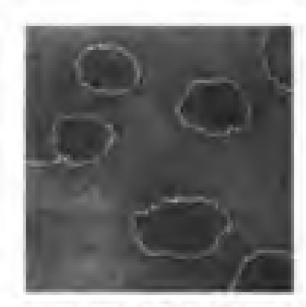

（d）将分水线叠加在原图上

图 10-33　分水岭分割算法

10.5.4　应用标记

直接使用分水岭分割算法，通常会由于噪声或其他诸如梯度的局部不规则性的影响造成过度分割。如图 10-34 所示，过度分割会使利用该算法得到的结果变得毫无用处，此时的过度分割意味着分割区域过多。一个较实际的解决方案是通过合并预处理步骤来限制允许存在的区域的数目，这些预处理步骤是为将附加知识用于分割过程而设置的。

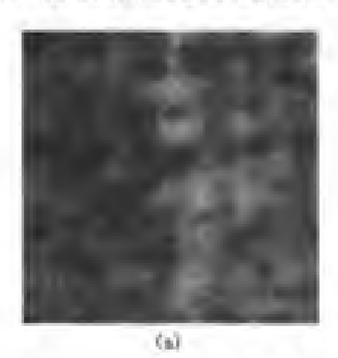

(a)　　(b)

图 10-34　过度分割

（a）原图像。

（b）对梯度图像使用分水岭分割算法得到的结果。

控制过度分割的方法是以标记的概念为基础的。一个标记是属于一幅图像的连通分量。这里有与重要对象相联系的内部标记，还有同背景相联系的外部标记。选择标记的典型过程包括两个主要步骤：

（1）预处理。

（2）定义一个所有标记必须满足的准则集合。

为了对此进行说明，再次考虑图 10-34（a）。导致图 10-34（b）中过度分割结果的一部分原因是大量隐含的最小值。由于这些区域的尺寸很小，所以这些最小值有很多是不相关的细节。在前面的讨论中，已不止一次的提到，将很小的细节对于图像的影响降至最低的有效方法是用一个平滑滤波器对图像进行过滤。在这种特殊情况下，这是一种合适的处理方案。

假设将内部标记限定为：

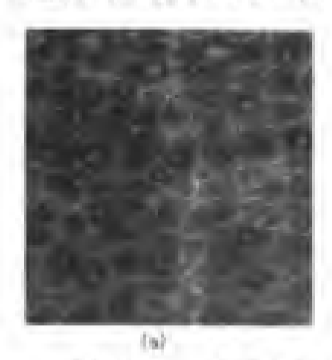
（1）被更高的“海拔”点包围起来的区域。

（2）区域中的点组成一个连通分量。

（3）所有属于这个连通分量的点具有相同的灰度值。

在图像经过平滑处理之后，满足这些限定的内部标记以图 10-35（a）中浅灰色、斑点状区域表示。下一步，对平滑处理后的图像使用分水岭算法，并限制这些内部标记只能是允许的局部最小值。图 10-35（a）显示了得到的分水线。将这些分水线定义为外部标记。沿着分水线的点是很好的背景候选点，因为它们经过相邻的标记之间的最高点。

图 10-35（a）中显示的外部标记有效地将图像分割成不同区域。每个区域包含一个惟一的内部标记和部分背景。但是这样就变成将每个这样的区域一分为二：单一的对象和它的背景，这个问题可应用多种在本章前面讨论过的分割技术来解决。另一种简单的方法是对每个单独的区域使用分水岭分割算法，即只求得平滑后的图像的梯度（如图 10-33（b）），然后约束算法只对包含特定区域中标记的分水岭进行操作。使用这种方法得到的结果显示于图 10-35（b）中，显然，相对图 10-34（b）有明显改善。


(a)

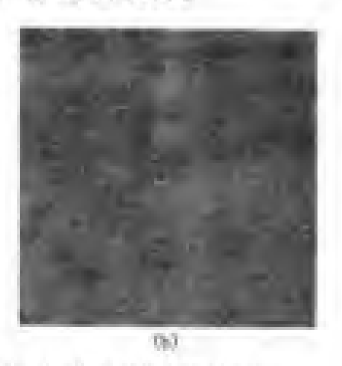

(b)

图 10-35　应用内部标记的分水岭分割算法

（a）显示有内部标记（浅灰色区域）和外部标记（分水线）的图像。

（b）分割的结果。

标记的选择可以用基于灰度值和连通性的简单的过程分类，如前面所说明的那样，更复杂的描述涉及尺寸、形状、位置、相对距离、纹理内容等等。关键是标记的使用给人们带来关于分割问题的某种带有先验性质的知识。实际上，人类经常以先验知识用日常的视觉辅助进行各种分割和更高级的工作，最熟悉的例子就是在阅读上下文时所用到的方法。因此，分水岭分割方法提出了一种能有效使用这类知识的机制，这是这种方法的一个突出优点。

10.6　运动在图像分割中的运用

当人类和动物要从不相关的背景中提取出所感兴趣的物体的时候，运动是可资利用的最有用的线索。在成像应用领域，运动是由于成像景物与传感器间的相对移动而产生的，比如：机器人的应用、自主导航和动态视觉分析等方面都是如此。

在下面的各节中，将从空间域和频率域两个方面来考虑运动在图像分割中的应用。

10.6.1　空域分割技术

1．基本方法

检测在 t_i 和 t_j 时刻所拍摄的两幅图像 $f(x,y,t_i)$ 和 $f(x,y,t_j)$ 之间的变化的一种最简单的

方法是将两幅图像的像素逐个进行对比。为进行上述的对比，可以计算出这两幅图像间的差值图像。假设有一幅静态参考图像，将这幅图像和随后生成的一幅处于同样环境中但包含一个运动物体的图像相比较，去掉两幅图像中的静态成分，余下的与非静态成分相对应的就是两幅图像的差值，即图像的运动成分。

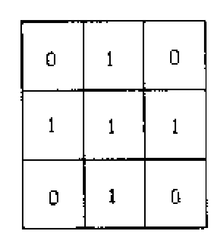
在t_i和t_j时刻两幅图像间的差值图像可以定义为：

$$d_{ij}(x,y)=\begin{cases}1 & \left|f(x,y,t_i)-f(x,y,t_j)\right|>\theta \\ 0 & \text{其他}\end{cases} \tag{10.6.1}$$

其中，θ是一个特定的阈值，当两幅图像在坐标(x,y)处的灰度值的差值大于阈值θ时，$d_{ij}(x,y)$才取值为 1，此处，假设所有的图像具有同样尺寸的大小。因为在式（10.6.1）中的坐标(x,y)跨越了这些图像的尺寸，所以差值图像$d_{ij}(x,y)$与上述这些序列中的图像大小相同。

在动态图像处理过程中，人们常将使得$d_{ij}(x,y)$的值为 1 的像素看成是物体运动的结果。但是，只有在所考虑的两幅图像都被记录而且两幅图像的照明要相对的保持恒定（其起伏要小于阈值θ）的情况下，这种方法才能适用。而在实际情况中，使得$d_{ij}(x,y)$的取值为 1 的项也常常可能是噪声。一般的，这些因为噪声而使得$d_{ij}(x,y)$取值为 1 的点在差值图像中表现为孤立的点。

去除这些孤立点的一个简单的方法是构造一个的 4-连通或 8-连通区域的空间模板（图 10-36），将它作用于差值图像，对于那些模板响应小于某一预定值且$d_{ij}(x,y)=1$的点就认为是由于噪声而引起的，应去除。



0	1	0
1	1	1
0	1	0

（a）4-邻域的空间模板

1	1	1
1	1	1
1	1	1

（b）8-邻域的空间模板

图 10-36 空间模板

虽然这种方法有可能忽略较小的或者运动缓慢的物体，但是它确实能在大多数情况下保留运动结果。

2. 累积差分

当试图从一个图像序列中将运动的部分分离出来时，差值图像中由于噪声而引起的孤立点并不是一个无足轻重的问题。虽然通过阈值连通性分析（Thresholded Connectivity Analysis）可以减少或者完全根除孤立点，但是这种滤波过程常常会导致较小的或者是缓慢运动的物体的丢失。处理这一问题的方法是考虑几帧图像中同一像素的变化，这样，就需要将“记忆”引入这一处理过程。这种方法包含的基本思想是忽略那些图像序列间偶尔出现的变化，并将其归于随机噪声的影响。

考虑一个图像序列$f(x,y,t_1)$，$f(x,y,t_2)$，…，$f(x,y,t_n)$，并令$f(x,y,t_1)$为基准图像。将基准图像与图像序列中的其他图像进行对比，可以得到一幅累积差分图像（ADI）。在每次比较中，如果基准图像和图像序列中的图像之间在某个像素位置上存在差异，则累积差

分图像中对应位置的计数器的计数加 1，这种计数器在累积差分图像的每个像素的位置都有一个。

因此，当第 n 帧图像与基准图像比较后，累积差分图像中的每一个像素值给出的是在该像素位置处，与基准图像灰度值存在差异的图像序列中的图像的数目。

图 10-37 详细描述了上述思想。图 10-37（a）~（e）表示的是一个长方形的物体（用灰度值 0 描述）向右以 1 像素/帧的速度匀速运动。图中给出的是位移 1 个像素后各瞬间的情况。图 10-37（a）给出的是基准图像，而图 10-37（b）~（e）给出的是图像中的第 2、3、4 帧以及第 11 帧图像。图 10-37（f）~（i）是对应的累积图像，它们可以这样解释：在图 10-37（f）中，左边取值为 1 的一列是取图 10-37（a）中的物体与图 10-37（b）中的背景间的差值的结果；右边取值为 1 的一列是由基准图像的背景与运动物体的前缘间的差异而引起的。在第 4 帧图像（如图 10-37（d）所示）所对应的时刻，累积差分图像中的第一个非零列为数字 3，它表示基准图像与图像序列中各帧图像在该列上共存在 3 个差值。最后，图 10-37（i）表示在该位置上共有 10（用十六进制中的"A"来表示）次变化。对于其他非零列可作类似讨论。

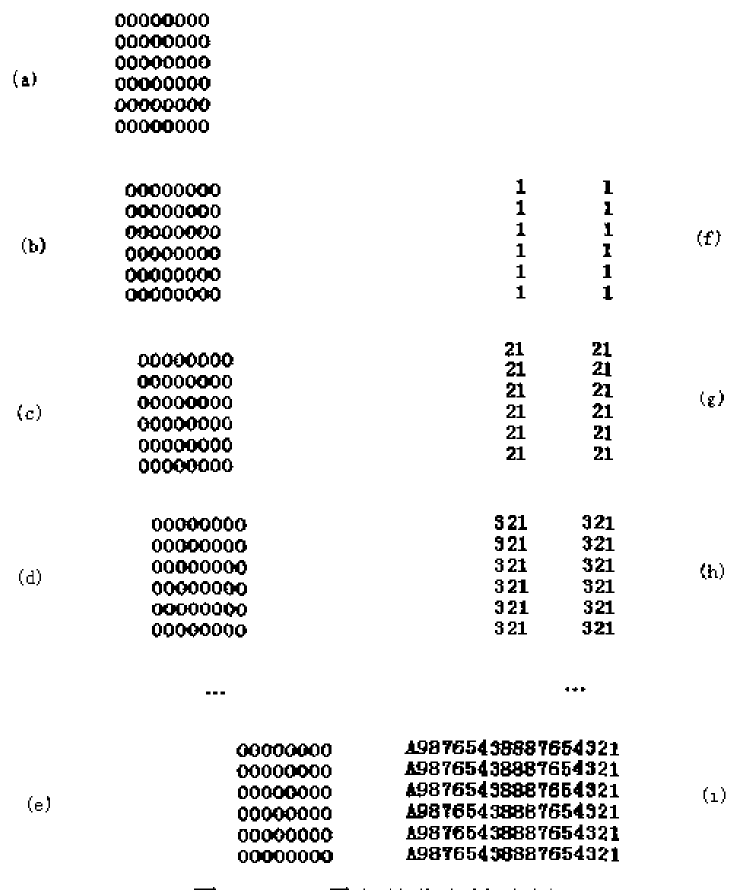


图 10-37 累积差分空域分割

通常，考虑下列三种类型的累积差分图像是有用的：绝对累积差分图像（AADI）、正累积差分图像（PADI）、负累积差分图像（NADI）。这里的正差分和负差分可以通过利用式（10.6.1）来计算。令 $R(x,y)$ 表示基准图像，为简化符号，令 k 表示 t_k，则有 $f(x,y,k)=f(x,y,t_k)$。假设 $R(x,y)=f(x,y,1)$，然后记住作为计数的 ADI 值，并对所有 $k>1$，对 (x,y) 的所有相关的值定义如下：

$$A_k(x,y)=\begin{cases}A_{k-1}(x,y)+1 & |R(x,y)-f(x,y,k)|>T\\ A_{k-1}(x,y) & \text{其他}\end{cases} \tag{10.6.2}$$

$$P_k(x,y)=\begin{cases}P_{k-1}(x,y)+1 & [R(x,y)-f(x,y,k)]>T\\ P_{k-1}(x,y) & \text{其他}\end{cases} \tag{10.6.3}$$

$$N_k(x,y)=\begin{cases}N_{k-1}(x,y)+1 & [R(x,y)-f(x,y,k)]<-T\\ N_{k-1}(x,y) & \text{其他}\end{cases} \tag{10.6.4}$$

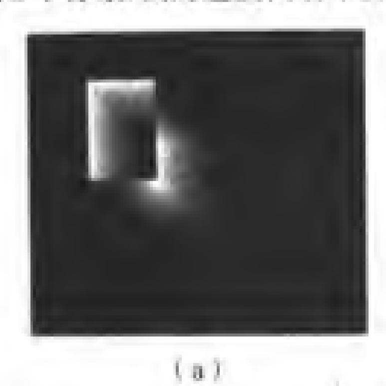
其中，$A_k(x,y)$、$P_k(x,y)$ 和 $N_k(x,y)$ 分别为与图像序列中的第 k 幅图像比较后的 AADI、PADI、NADI。

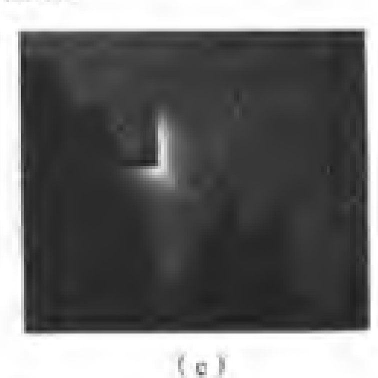
不难理解，这些 ADI 是从所有计数值为零时开始，且 ADI 同序列中的图像大小相同。如前面提到的，序列中的图像都被假设为同样大小。

注意：如果背景像素的灰度值大于运动物体的灰度值，则式（10.6.2）和式（10.6.3）中的不等式的顺序以及阈值的符号都是相反的。

【例 10-9】累积差分图像示例。

图 10-38 显示了以三个灰度图像显示的 ADI，图中的矩形物体大小为 75×50 像素，以 $5\sqrt{2}$ 个像素/帧的速度向右下方移动。图像大小为 256×256 像素。





（a）（b）（c）

图 10-38 向右下方运动的矩形目标

（a）AADI。

（b）PADI。

（c）NADI。

从图中，明显可以看到：

（1）PADI 的非零区域等于运动对象的大小。

（2）PADI 的位置对应于基准图像中运动对象的位置。

（3）PADI 停止计数时给出的位置，即为物体在基准图像中的位置。

（4）AADI 包括 PADI 的和 NADI。

（5）运动对象的速度和方向可以由 AADI 和 NADI 决定。

3. 基准图像的设置

从上面的讨论可知，找到一幅可以与后续图像进行比较的基准图像是运用此技术的关键，只有找到一幅基准图像，才可以将图像序列中的其他图像与该图像进行比较。正如前面提到的那样，在动态成像问题中，两幅图像间的差分运算将消除图像中静态的成分，而只留下相应于噪声和运动物体的成分。有关噪声，可以通过滤波方法及构造一个累积差分

图像的方法来解决。

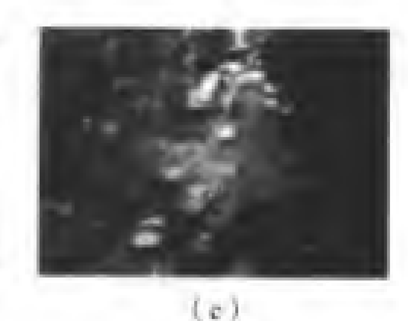
实际上，要获得一幅只含有静态成分的基准图像往往是不可能的，因此，有必要研究如何从包含一个或几个运动物体的图像序列中构造出基准图像。可以按照下述方法生成基准图像：将图像序列中的第一帧图像作为基准图像，当一个非静态成分完全移出了它在参考图像中的位置，则该非静态成分在当前一帧图像中的背景可以被复制到参考图像中该运动物体原来所占据的位置。当所有的物体都完全移出了它们原来所处的位置的时候，就可以得到一幅只含有静态成分的参考图像。

【例 10-10】构造基准图像。

图 10-39（a）和图 10-39（b）是两幅关于一个十字路口的图像帧。把第一幅图像作为基准图像，第二幅是一段时间后的同一场景。目的是去掉基准图像中主要的运动对象，生成一幅静态图像。尽管还有其他更小的运动对象，但主要的运动特征是十字路口上从左到右运动的汽车。为了说明，着重观察这个对象。通过监视正的 ADI 的变化，可以如前面所讨论的那样确定运动物体的初始位置。一旦确定这个区域被此对象所占据，可以通过剪影法将对象从图像中除去。通过观察图像序列中 PADI 停止变化的那一帧图像，可以从这幅图像中复制在初始帧中原来由运动对象占据的区域。然后这个区域就被贴到被挖去对象的图像中，从而恢复这个区域的背景。如果对所有运动的对象都进行这样的操作，结果正如前面解释的那样，可以得到一幅可以和后续帧进行比较、对于运动检测只包含静态图像部分的基准图像。图 10-39（c）中显示了去掉向东运动的汽车，并根据图 10-39（b）中对应的区域恢复背景得到的结果。



（a）　（b）



（c）

图 10-39　构造一幅基准图像

（a）、（b）为图像序列中的两帧图像。

（c）从（a）中去掉向右运动的汽车并根据（b）中对应的区域恢复背景。

10.6.2　频域技术

在这一部分，将考虑用傅立叶变换公式进行运动估计。T 帧由静态照相机所拍摄的尺寸为 M×N 的图像序列，$t=0,1,\cdots,T\text{-}1$。假设图像序列中的每帧图像都具有均匀的背景且背景的强度为 0。在这一系列图像中，有一个大小为 1 像素、强度为 1 的物体匀速运动。对第一帧（t＝0）图像，将它投影到 x 轴上，即沿着图像的每一列对像素的强度求和。经过这样的操作，会得到一个具有 M 个分量的一维数组。除了运动物体投影的位置外，该一维数组的其他分量都为 0。

如果将上述数组的每个分量都乘以 $\exp(j2\pi k_1 x\Delta t)$，$x=0,1,\cdots,M-1$，对每一个分量求和，可得到 $\exp(j2\pi k_1 x\Delta t)$，其中 (x',y') 为运动物体在该时刻的瞬间位置。此处，k_1 是

一个正整数，Δt 是每帧图像间的时间间隔。

假设在第二帧中（$t=1$），对象运动到坐标 $(x'+1,y')$，即对象在沿 x 轴方向上移动了一个像素，重复上面的投影过程，可得到和为 $\exp(j2\pi k_1(x'+1)\Delta t)$。如果物体继续以 1 像素/帧的速度运动，则在任一整时刻，都可以得到这样的结果：

$\exp(j2\pi k_1(x'+t)\Delta t)$，

利用欧拉公式，可以表达为：

$$\exp[j2\pi k_1(x'+t)\Delta t]=\cos[2\pi k_1(x'+t)\Delta t]+j\sin[2\pi k_1(x'+t)\Delta t] \quad (10.6.5)$$

其中，$t=0,1,\cdots$，T-1。可以看到，前面讨论的步骤给出了一个复正弦曲线，其频率为 k_1。如果物体在相邻帧图像间位移了 v_1 像素，则正弦曲线的频率为 v_1k_1。因为 t 是以整数增量在区间[0,T-1]上变化的，所以将频率限制为整数，就可以得到复正弦曲线的离散傅立叶变换，它在频率 v_1k_1 和 $T-v_1k_1$ 处有两个峰值，其中后一个峰为对称性折叠的结果，可忽略。所以，如果对傅立叶频谱进行峰值搜索，可以得到频率 v_1k_1。如果每帧图像间的切换频率 k_1 已知，将上面所得到的频率除以 k_1 就可以得到 v_1，也即物体沿 x 轴方向运动的速率。如果将图像投影到 y 轴，同理可得物体沿 y 轴方向运动的速率 v_2。

对于一个没有运动发生的图像序列，可以得到完全一致的指数项，它的傅立叶变换谱只在频率为零的位置存在一个峰（单一的直流峰（DC 峰））。由于前面所讨论的各变换都是线性的，所以对于一般的在静止背景上包含多个物体运动的情况，其傅立叶变换也一定含有一个相应于静态图像成分的直流峰以及相应于每个运动物体的峰值，其位置正比于该物体的运动速度。

上述的概念可总结成以下的表达式。对于一个包含 T 帧大小为 M×N 的数字图像序列，分别求它们在 x 轴上的加权投影和，得到：

$$g_x(t,k_1)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y,t)\exp(j2\pi k_1x\Delta t) \quad t=0,1,\cdots,\text{T-1} \quad (10.6.6)$$

类似的，图像在 y 轴上的投影的加权和可以表达为：

$$g_y(t,k_2)=\sum_{y=0}^{N-1}\sum_{x=0}^{M-1}f(x,y,t)\exp(j2\pi k_2y\Delta t) \quad t=0,1,\cdots,\text{T-1} \quad (10.6.7)$$

其中，k_1、k_2 为正整数。

式（10.6.5）和式（10.6.6）的一维傅立叶变换可分别表示为：

$$G_x(u_1,k_1)=\frac{1}{T}\sum_{t=0}^{T-1}g_x(t,k_1)\exp(-\frac{j2\pi u_1t}{T}) \quad u_1=0,1,\cdots,K-1 \quad (10.6.8)$$

和：

$$G_y(u_2,k_2)=\frac{1}{T}\sum_{t=0}^{T-1}g_y(t,k_2)\exp(-\frac{j2\pi u_2t}{T}) \quad u_2=0,1,\cdots,K-1 \quad (10.6.9)$$

在实际应用中，上述变换通常采用快速傅立叶变换 FFT 来实现。在这里，频率—速度关系为：

$$u_1=k_1v_1 \quad (10.6.10)$$

和：

$$u_2=k_2v_2 \quad (10.6.11)$$

在这两个公式中，速度的单位是“像素/总帧时间”。例如：$v_1=10$，表示物体在 T 帧

图像对应的时间间隔内位移了 10 像素。

如果按均匀的时间间隔拍摄每帧图像，则实际的物理速度依赖于帧变换率以及像素之间的距离。因此，如果 $v_1 = 10$，T = 30，帧变换率为 2 帧/s，像素间的实际距离为 0.5m，则 x 轴方向的实际物理速度为：

$$v_1 = 10\text{像素} \times 0.5\text{m}/\text{像素} \times \frac{2\text{帧}/s}{30\text{帧}} = \frac{1}{3} m/s$$

物体运动速度的 x 分量的符号，可以通过如下的方式来计算：

$$S_{1_x} = \left| \frac{d^2 \operatorname{Re}[g_x(t,k_1)]}{dt^2} \right|_{t=n} \tag{10.6.12}$$

和：

$$S_{2_x} = \left| \frac{d^2 \operatorname{Im}[g_x(t,k_1)]}{dt^2} \right|_{t=n} \tag{10.6.13}$$

由于 g_x 是正弦曲线，可以证明，如果速度分量 v_1 为正值，在时刻 n，S_{1x} 和 S_{2x} 在任意位置具有相同的符号。反之，如果 S_{1_x} 和 S_{2_x} 具有相反的符号，则标明速度分量 v_1 的符号为负。如果 S_{1_x} 和 S_{2_x} 中任意一个为零，则考虑位于时刻 $t = n \pm \Delta t$ 的下一个最近的点。同理，可以计算出 v_2 的符号。

【例 10-11】 通过频域检测小的运动目标。

图 10-40 显示了一个包含 32 帧对地观测卫星图像序列中的一张，这组图像的参考图像中加入了白噪声图像，且图像序列中叠加了一个运动目标，这个目标在 x 轴方向上的速度为 0.5 像素/帧，在 y 轴上的速度为 1 像素/帧。在一个小区域内（9 像素），目标具有高斯灰度分布，且用肉眼不易辨别。

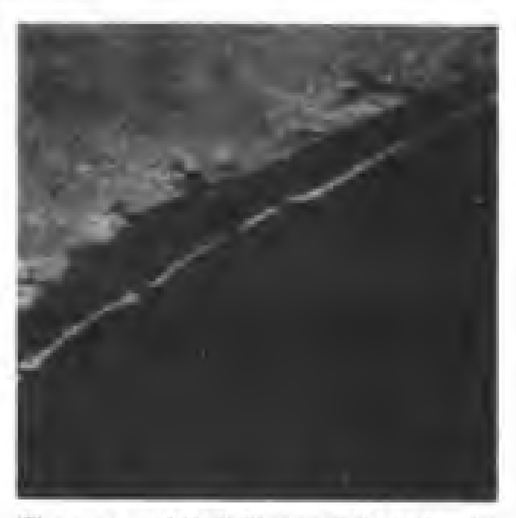


图 10-40　对地观测卫星图像中的一帧

令 $k_1 = 6$，$k_2 = 4$，计算式（10.6.7）和式（10.6.8）得到的结果分别显示于图 10-41 和图 10-42。图 10-41 中位于 $u_1 = 3$ 处的波峰根据式（10.6.9）得到 $v_1 = 0.5$。同样，图 10-42 中的波峰根据式（10.6.10）得到 $v_2 = 1.0$。可以根据图 10-41 和图 10-42 来选择 k_1 和 k_2。例如，假设令 $k_2 = 15$，此时，图 10-42 中的波峰将位于 $u_2 = 15$ 和 $u_2 = 17$。因为，$v_2 = 1.0$，这会导致一个严重混淆的结果。混淆是由于采样过疏造成的（由于 u 的范围由 T 决定，而当前的讨论中使用的帧太少）。因为 $u = kv$，一种可能性是选择 k 作为最接近 $k = u_{max} / v_{max}$ 的

整数，这里u_{max}是确定T的混淆率限制，而v_{max}是对象的速度最大值。

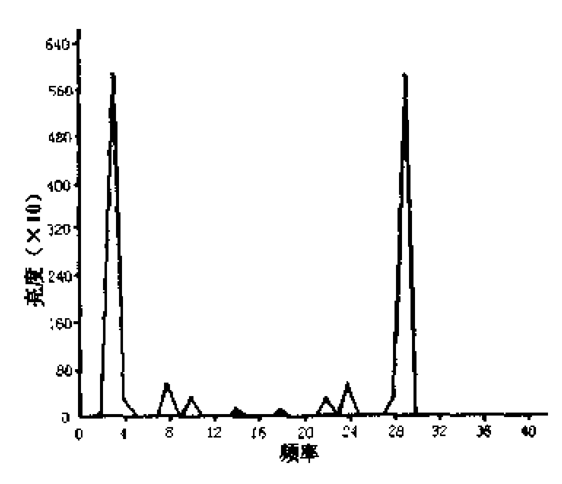


图10-41 式（10.6.7）的频谱，显示位于$u_1=3$处的一个波峰

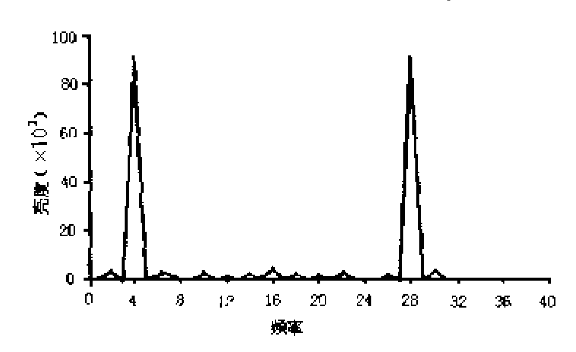


图10-42 式（10.6.8）的频谱，显示位于$u_2=4$处的一个波峰

小结

所谓分割，是按一定的规则将一幅图像或景物分成若干个部分或子集的过程。图像分割的基本概念是将有意义的特征或者需要应用的特征提取出来。图像分割大致可分为相似性分割和连续性分割。本章介绍了常用的分割算法，并以实例详细说明，包括阈值分割、区域增长、形态学分水岭分割等。

重点：本章的重点是边缘连接和边界检测；阈值处理，包括全局阈值处理、自适应阈值处理等；基于区域的分割，包括区域生长、区域分裂与合并基于形态学分水岭的分割。

难点：基于形态学的分水岭算法的原理和应用。

疑点：图像复原技术的发展已经超出了当初的目标，如何在图像处理中应用图像复原技术以实现更多的应用层次。

练习十

一、选择题

1. 图像的三种基本类型的不连续性是（ ）。

   A. 点　　B. 线　　C. 边缘　　D. 梯度

2. 采用模板$[-1\ \ 1]'$主要检测（ ）方向的边缘。

A. 水平　　B. 45°　　C. 垂直　　D. 135°

3. 列算子中利用边缘灰度变化的二阶导数特性检测边缘的是（　）。

A. 梯度算子　　B. Prewitt 算子　　C. Roberts 算子　　D. Laplacian 算子

二、思考题

1. 有一幅包含水平的、垂直的、+45°和−45°直线的二值图像。试给出一组 3×3 模板可以用于检测这些直线中一个像素宽度的间断。假设直线的灰度级是 1 并且背景的灰度级是 0。

2. 一幅图像背景部分的均值为 25，方差为 625，在背景上分布着一些互不重叠的均值为 150、方差为 400 的小目标。设所有目标合起来约占图像总面积的 20%，试提出一个基于阈值的分割算法将这些目标分割出来。

3. 在显示的子图中找到对应于最小开销路径的边缘。括号中的数字是灰度级，外部的数字是空间坐标。如图 10-43 所示，假设边缘起始于第一列而结束于最后一列。

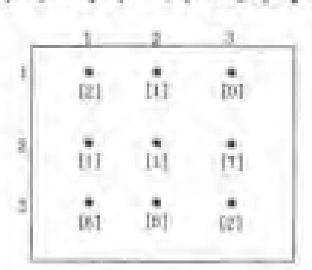


图 10-43　图像

4. 假设图像的灰度级概率密度函数如图 10-44 所示。这里 $p_1(z)$ 对应于图像，$p_2(z)$ 对应于背景。假设 $P_1 = P_2$，找出对象像素和背景像素之间的最佳门限。

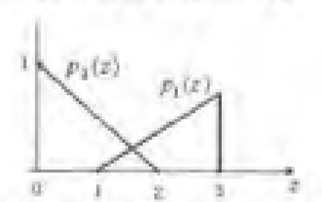


图 10-44　灰度级概率密度函数

5. 用分裂合并法分割图 10-45，并给出对应分割结果的四叉树。

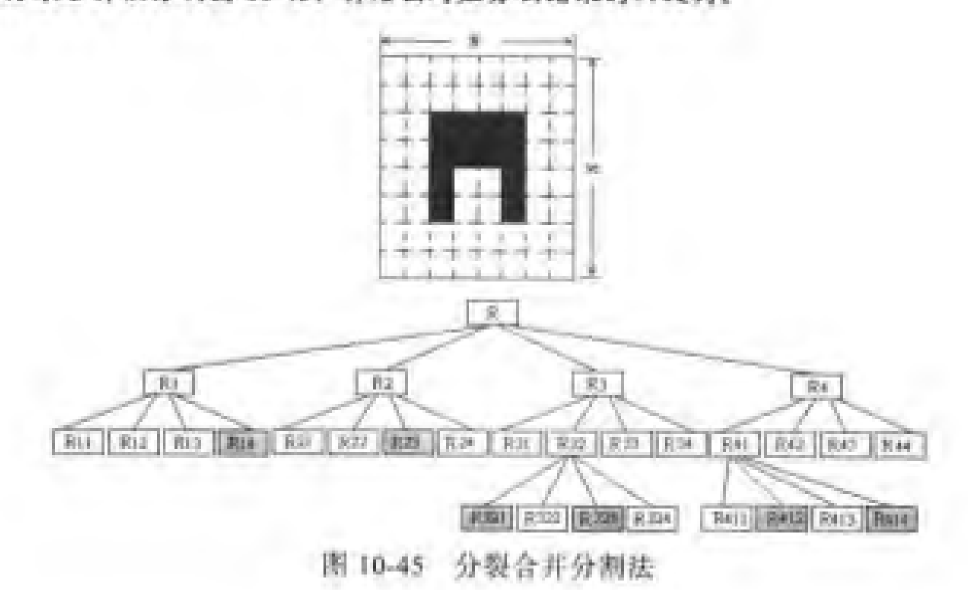


图 10-45　分裂合并分割法

三、上机题

1. 编程实现用 Sobel 算子、Prewitt 算子、Laplacian 算子、LOG 算子分别对图 10-46 进行边缘检测，并比较边缘图的有何不同。对原图像加入不同噪声后再分别进行边缘检测，观察边缘提取效果是否有所不同。



图 10-46 图像

2. 编程实现对图 10-26(a)基于全局阈值的分割，取不同的阈值，观察分割结果；根据本章中提到的自适应阈值计算方法计算阈值，观察分割结果。

第 11 章　基于形态学的图像处理

数学形态学（Mathematical Morphology）源自于 Minkowski 代数。1964 年 Matheron 与 serra 提出时，只适用于二元图像（binary image）。后经 Serra 与 Sternberg 等人的努力，将数学形态学扩充至灰度值图像。

形态学为大量的图像处理问题提供了一种一致而有力的方法。数学形态学的语言是集合论，其中所提及的集合，表示图像中的不同对象。例如，在二值图像中，所有白色像素的集合是图像完整的形态学描述。

本章的主要内容有以下几个方面：

（1）基本概念

（2）膨胀与腐蚀。

（3）开启与闭合。

（4）击中与击不中变换。

（5）一些基本的形态学算法。

（6）灰度级图像扩展。

11.1　基本概念

11.1.1　集合论的几个基本概念

令 A 为一个 Z 中的集合。如果 $a=(a_1,a_2)$ 是 A 的元素，则记为：

$$a \in A \qquad (11.1.1)$$

否则，记为：

$$a \notin A \qquad (11.1.2)$$

不包含任何元素的集合称为空集，用符号 ϕ 表示。

集合由两个括号之中的内容表示：{◎}。本章关注的集合元素是图像中描述的对象或其他感兴趣特征的像素坐标。

例如，形如 $A=\{a \mid a=-b, b \in B\}$ 的表达式意思为：集合 A 是元素 a 的集合，而 a 是通过 −1 与集合 B 中的所有元素的两个坐标相乘得到的。

如果集合 A 中的所有元素都属于集合 B，则称 A 为 B 的子集，记为：

$$A \subseteq B \qquad (11.1.3)$$

给定两个集合 A，B 如图 11-1 所示。

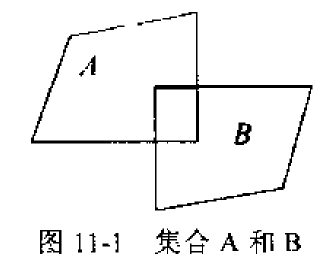


图 11-1　集合 A 和 B

则它们可以进行以下运算，运算结果用灰色表示：

两个集合的并集记为：

$$C = A \bigcup B = \{c \mid c \in A or c \in B\} \tag{11.1.4}$$

如图 11-2 所示。

两个集合的交集记为：

$$D = A \bigcap B = \{d \mid d \in A \& d \in B\} \tag{11.1.5}$$

如图 11-3 所示。

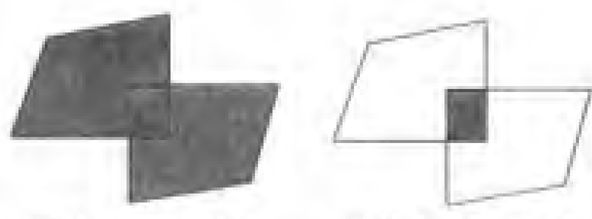


图 11-2 $C = A \bigcup B$　　图 11-3 $D = A \bigcap B$

如果两个集合没有共同元素，则称为不相容或者互斥，记为：

$$A \bigcap B = \varnothing \tag{11.1.6}$$

集合 A 的补集是不包含于 A 的所有元素的集合，记为：

$$A^c = \{w \mid w \notin A\} \tag{11.1.7}$$

如图 11-4 所示。

集合 A 和 B 的差，表示为 A－B，定义为：

$$A - B = \{w \mid w \in A, w \notin B\} = A \bigcap B^c \tag{11.1.8}$$

如图 11-5 所示。

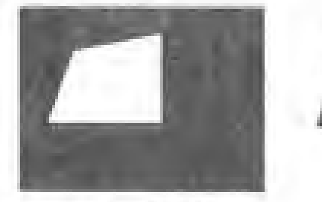


图 11-4　A^c

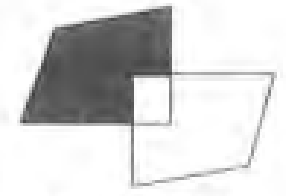


图 11-5　A-B

还需要另外两个能广泛应用于形态学的附加定义。

集合 A 平移到 $z = (z_1, z_2)$，表示为（A）z，定义为：

$$(A)_z = \{w \mid w = a + z, a \in A\} \tag{11.1.9}$$

如图 11-6 所示。

集合 A 的反射，表示为 $\hat{A}$，定义为：

$$\hat{A} = \{w \mid w = -a, a \in A\} \tag{11.1.10}$$

如图 11-7 所示。

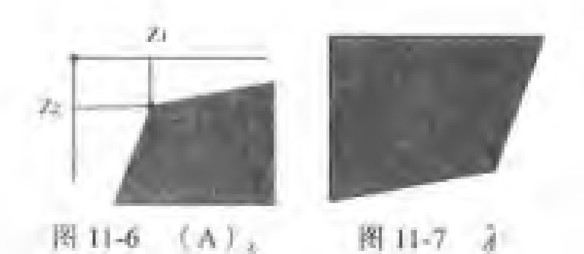


图 11-6　（A）$_z$　　图 11-7　$\hat{A}$

11.1.2　二值图像的逻辑运算

逻辑运算对于实现以形态学为基础的图像处理算法是一种有利的补充手段。下面讨论涉及二值像素和图像的逻辑运算。

在图像处理中主要用到的逻辑运算：与、或和非。表 11-1 中总结了这些运算的法则。这些运算可以互相组合形成其他逻辑运算。

表 11-1　逻辑运算的法则

p	q	p 与 q	p 或 q	非 p
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

在两幅或多幅图像的对应像素间对像素进行逻辑运算（除了“非”运算，此运算只对单一图像的像素进行）。因为只有在两个变量都是 1 时，两个二进制变量的“与”运算才为 1，所以只有两幅输入图像的对应像素均为 1 时，“与”运算的结果图像任何位置的结果才是 1。使用表 11-1 中的定义构造其他的逻辑运算很容易。例如，异或运算是当两个像素的值不同时结果为 1，否则为 0。这种运算与“或”运算不同。“或”运算在两个像数有一个为 1 或两个均为 1 时结果为 1。同样，非—与运算可选出在 B 中而不在 A 中的 1 像素。

注意：刚才讨论的逻辑运算与 9.1.1 节中讨论的集合运算有一一对应的关系是很重要的，并且逻辑运算被限制只对二进制变量进行运算，而这通常不是集合运算所处理的情况。因此，比如集合论中的交集运算在运算对象为二进制变量时，归为“与”运算。“相交”和“与”这类术语（甚至它们的符号）经常在各种著述中交替地用于表示一般的或二进制值的集合运算。通常从讨论的上下文中可以清楚地知道它们的意义。

11.2　膨胀与腐蚀

膨胀与腐蚀是形态学处理的基础，本章中讨论的许多形态学算法都是以这两种原始算法为基础的。

二值形态学中的运算对象是集合，也就是二值矩阵，实际上当涉及两个二值矩阵时，并不把它们对等看待。通常设 A 为图像矩阵，B 为结构元素矩阵，数学形态学运算时，用 B 对 A 进行操作。实际上，结构元素本身也是一个图像矩阵。这里对每个结构元素矩阵指定一个原点。

11.2.1　膨胀

由于 A 和 B 是 Z^2 中的集合，A 被 B 膨胀的定义为：

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \varnothing\} \qquad (11.2.1)$$

这个公式表示用 B 膨胀 A 的过程是，相对 B 做关于中心像素的映射，在将其映射平移 x，这里 A 与 B 映像的交集不为空集。式（11.2.1）可以重写为：

$$A \oplus B = \{z \mid [(\hat{B})_z \cap A] \subseteq A\} \qquad (11.2.2)$$

上式可以帮助人们借助卷积的概念来理解膨胀操作。如果将 B 看作是一个卷积模板，

膨胀就是先对 B 做关于中心像素的映射，再将映射连续地在 A 上移动而实现的。

图 11-8（a）显示了一个简单的集合，图 11-8（b）显示了一个结构元素和它的中心像素，图 11-8（c）显示了用这个结构元素膨胀的结果。结果显示，膨胀运算将原图像区域扩大了。

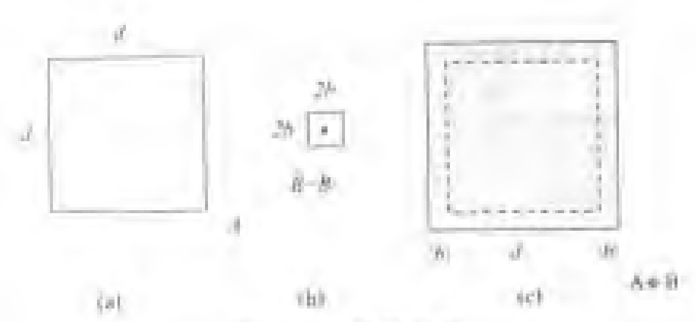


图 11-8 膨胀运算

示例程序 11-1 在 MATLAB 中，用 dilate 函数来实现膨胀操作。

【示例程序 11-1】

%采用以下结构要素矩阵

$$SE=\begin{bmatrix}1&1&1&1\\1&1&1&1\\1&1&1&1\\1&1&1&1\end{bmatrix}$$

```
%读入 cameraman.tif 并将其二值化后进行膨胀操作
I=imread('cameraman.tif');
figure, imshow(I);
J=im2bw(I);
figure,imshow(J);
SE=ones(4,4)
BW1=dilate(J,SE);
figure, imshow(BW1);
```

如图 11-9 所示，为膨胀前后的结果对比。




图 11-9 膨胀前后的结果对比

11.2.2 腐蚀

对 Z^2 中的集合 A 和 B，用 B 腐蚀 A 的定义为：

$$A\Theta B=\{z\mid(B)_z\subseteq A\} \qquad (11.2.3)$$

这个公式说明，使用 B 对 A 进行腐蚀是所有 B 中包含于 A 中的点 z 的集合用 z 平移。

图 11-10 显示了与图 11-8 相似的过程。图 11-10（a）显示了一个简单的集合，图 11-10（b）显示了一个机构元素和它的中心像素，图 11-10（c）显示了用这个结构元素腐蚀的结果。对比图 11-8（c）以前的集合 A 显示为虚线。阴影区域的边界说明了 B 的中心像数进一步移动的界限。结果显示，腐蚀运算将原图像区域缩小了，就像剥了一层皮一样，这也是其得名的原因。

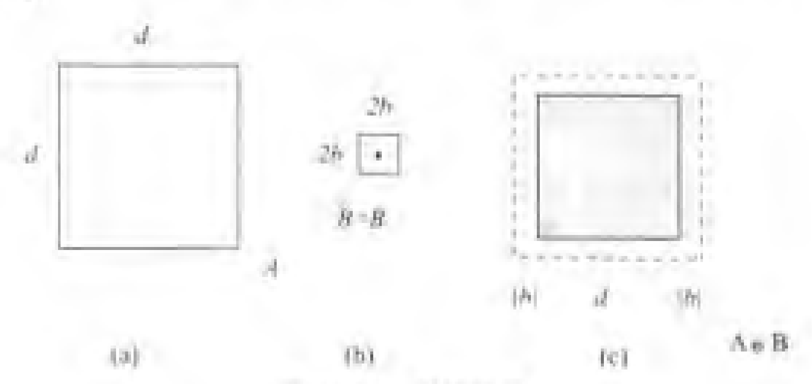


图 11-10　腐蚀运算

示例程序 11-2 在 MATLAB 中用 erode 函数来实现腐蚀操作。

【示例程序 11-2】

%采用以下结构要素矩阵

$$SE=\begin{bmatrix}1&1&1&1\\1&1&1&1\\1&1&1&1\\1&1&1&1\end{bmatrix}$$

```
%读入 cameraman.tif 并将其二值化后进行腐蚀操作
I=imread('cameraman.tif');
figure, imshow(I);
J=im2bw(I);
figure,imshow(J);
SE=ones(4,4)
BW1=erode(J,SE);
figure, imshow(BW1);
```

图 11-11 所示为腐蚀前后的结果对比。

图 11-11　腐蚀前后的结果对比

11.2.3　膨胀与腐蚀的对偶性

膨胀与腐蚀对于集合求补运算和反射运算是彼此对偶的。即：

$(A \Theta B)^c = A^c \oplus \hat{B}$ （11.2.4）

为了说明确定形态学表达式有效性的典型方法，需对这个结果进行正规的证明。从腐蚀的定义开始，有：

$A \Theta B = \{z \mid (B)_z \subseteq A\}$

如果集合 $(B)_z$ 包含于集合 A，则 $(B)_z \cap A^c = \varnothing$，此时前述公式变为：

$A \Theta B = \{z \mid (B)_z \cap A^c = \varnothing\}^c$

但满足 $(B)_z \cap A^c = \varnothing$ 的 z 的集合的补集是满足 $(B)_z \cap A^c \neq \varnothing$ 的集合。

因此：

$A \Theta B = \{z \mid (B)_z \cap A^c \neq \varnothing\} = A^c \oplus \hat{B}$

最后一步是从式（11.2.1）中得到的。这就证明了这个结论。

11.3 开启与闭合

本节讨论另外两个重要的形态学操作：开启与闭合。如前所述，膨胀使图像扩大而腐蚀使图像缩小，由于膨胀与腐蚀并不是互为逆运算，所以可以将它们级联结合使用。开启就是相对图像进行腐蚀，然后膨胀其结果。闭合就是先对图像进行膨胀，然后腐蚀其结果。开启一般使对象的轮廓变得光滑，断开狭窄的间断和消除细的突出物。闭合同样使轮廓线变得光滑，但与开启相反的是，它通常消除狭窄的间断和长细的鸿沟，消除小的孔洞，并填补轮廓线中的断裂开启。

11.3.1 开启

使用结构元素 B 对集合 A 进行开启操作，表示为 $A \circ B$，定义为：

$A \circ B = (A \Theta B) \oplus B$ （11.3.1）

开启有一个简单的集合解释（图 11-12）。假设将结构元素 B 看成一个转动圆盘，$A \circ B$ 的边界通过 B 中的点完成，即 B 在 A 中的边界内转动时，B 中的点所能到达的 A 的边界的最远点。这个开启的几何拟合特性使人们得出了集合论的一个公式。这个公式说明开启是通过求取 B 在拟合 A 时的平移的并集得到的。

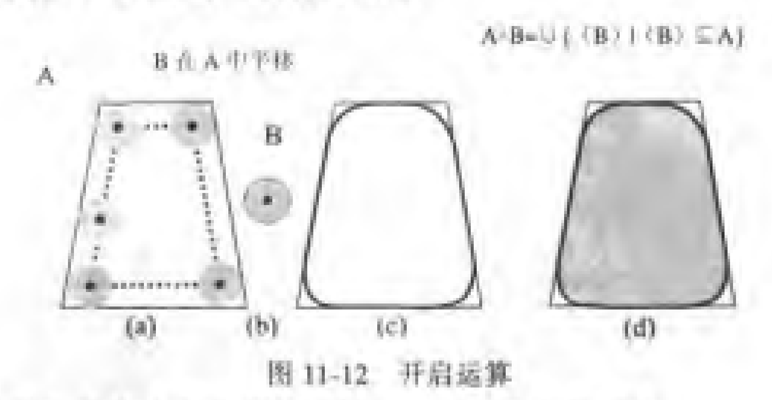


图 11-12 开启运算

（a）结构元素 B 沿着 A 的内部边界转动，点表示 B 的圆心。

（b）结构元素 B。

（c）粗线时开启操作的外部边界。

（d）灰色部分为开启的结果。

就是说，开启可以表示为：

$$A \circ B = \bigcup\{(B)_z \mid (B)_z \subset A\} \tag{11.3.2}$$

示例程序 11-3 根据定义对 circlesm.tif 用 30×30 的全 1 矩阵进行开启操作，结果如图 11-13 所示。

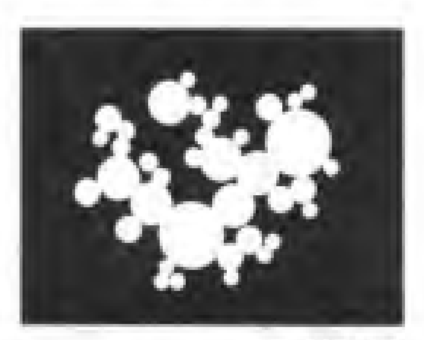
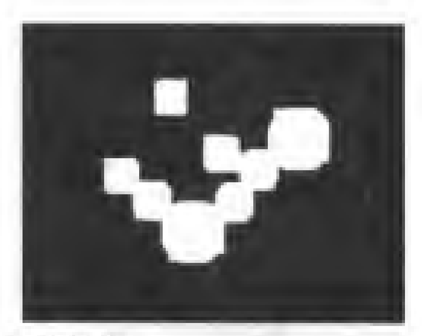



图 11-13　开启操作

【示例程序 11-3】

```
%读入并将其进行开启操作
I = imread('circlesm.tif');
figure,imshow(I);
SE = ones(30,30)
BW1 = erode(I,SE);
BW2 = dilate(BW1,SE);
figure, imshow(BW2);
```

也可用 MATLAB 提供的 BW1 = bwmorph（I, 'open'）函数，如示例程序 11-4。此时结构函数被固定为 3×3 全 1 矩阵，基本看不出图像处理前后的差别。这是由于结构函数的不同，所得的结果，相差比较大，如图 11-14 所示。所以，在需要调整结构函数的情况下，可以根据定义进行开启。

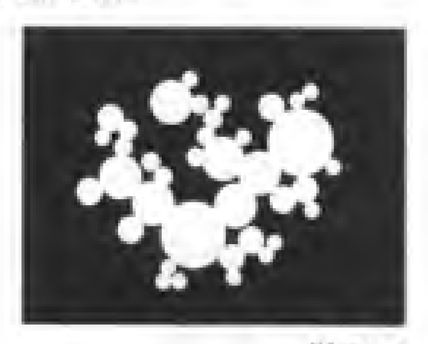
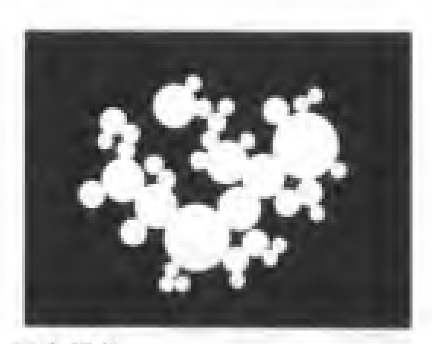



图 11-14　开启操作

【示例程序 11-4】

```
I = imread('circlesm.tif');
figure,imshow(I);
BW1 = bwmorph(I,'open');
figure,imshow(BW1);
```

11.3.2　闭合

使用结构元素 B 对集合 A 进行闭合操作，表示为 $A \bullet B$，定义为：

$$A \bullet B = (A \oplus B) \ominus B \tag{11.3.3}$$

类似于开启，闭合也有相似的几何解释，只是现在在边界的外部转动 B。如图 11-15

所示。

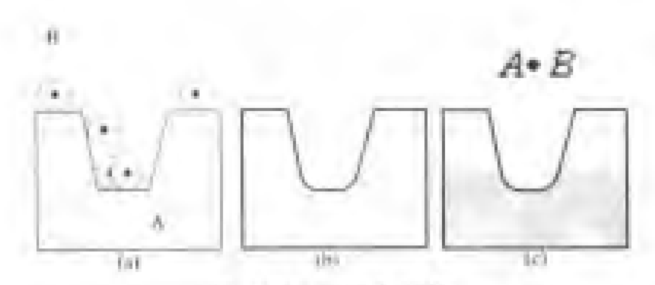


图 11-15　闭合运算

示例程序 11-5 根据定义对 circlesm.tif 用 30×30 的全 1 矩阵进行闭合作用，结果如图 11-16 所示。

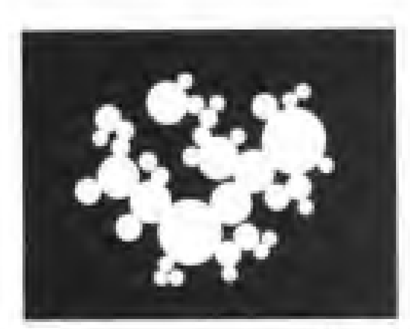
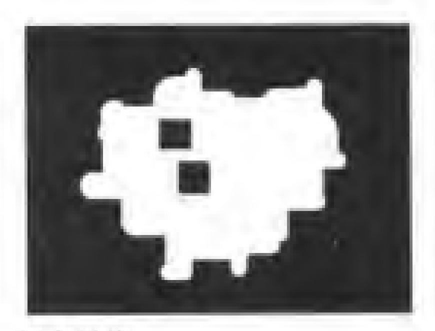



图 11-16　闭合操作

【示例程序 11-5】

```
%读入并将其进行闭合操作
I=imread('circlesm.tif');
figure,imshow(I);
SE=ones(30,30)
BW1=dilate(I,SE);
BW2=erode(BW1,SE);
figure, imshow(BW2);
```

也可用 MATLAB 提供的 BW1 = bwmorph（I，'close'）函数，如示例程序 11-6。此时结构函数被固定为 3×3 全 1 矩阵，如前面所讲由于结构函数的不同，所得的结果，相差比较大，如图 11-17 所示。所以，在需要调整结构函数的情况下，可以根据定义进行闭合。

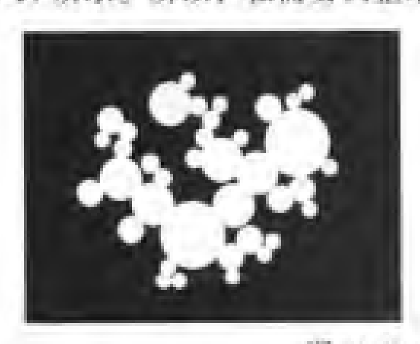
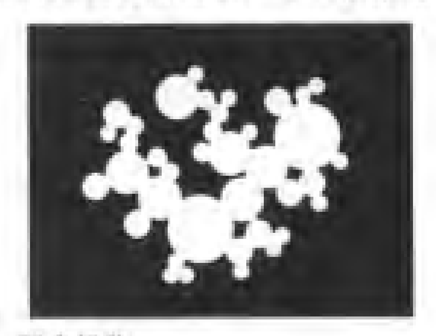



图 11-17　闭合操作

【示例程序 11-6】

```
I=imread('circlesm.tif');
figure,imshow(I);
BW1=bwmorph(I,'close');
figure,imshow(BW1);
```

11.3.3 开启与闭合的对偶性

由上可见，开启与闭合是一对对偶操作，如膨胀与腐蚀一样，开启与闭合也是一对关于集合求补以及映像的对偶操作，即：

$$(A \bullet B)^c = (A^c \circ \hat{B}) \qquad (11.3.4)$$

开启满足以下性质：

（1）$A \circ B$ 是 A 的子集合。

（2）如果 C 是 D 的子集，则 $C \circ B$ 是 $D \circ B$ 的子集。

（3）$(A \circ B) \circ B = A \circ B$

同样，闭合也满足下列性质：

（1）A 是 $A \bullet B$ 的子集。

（2）如果 C 是 D 的子集，则 $C \bullet B$ 是 $D \bullet B$ 的子集。

（3）$(A \bullet B) \bullet B = A \bullet B$

注意：由两种情况下的性质（3）可知，算子应用一次后，一个集合进行多少次开启和闭合操作都不会有变化。

11.4 击中与击不中变换

形态学上的击中与击不中变换是形状检测的基本工具。设 X 位于一个小窗口 W 中。与 W 有关的 X 的局部背景定义为集合的差（W－X）。如果 B 表示由 X 和 X 的背景构成的集合，则在 A 中对 B 进行的匹配表示为 $A \circledast B$：

$$A \circledast B = (A \Theta X) \cap [A^c \Theta (W - X)] \qquad (11.4.1)$$

可以通过令 $B = (B_1, B_2)$ 对这种表示法稍微进行推广。这里 B_1 是由与一个对象相联系的 B 元素构成的集合。B_2 是与响应背景有关的 B 元素的集合。根据前面的讨论 $B_1 = X$，$B_2 = W - X$。用这个方法，式（11.4.1）变为：

$$A \circledast B = (A \Theta B_1) \cap [A^c \Theta B_2] \qquad (11.4.2)$$

因此集合 $A \circledast B$ 同时包含了所有的原点，B_1 在 A 内找到匹配，B_2 在 A^c 内找到匹配。通过应用式（11.1.8）给出的集合之差的定义和式（11.2.4）给出的腐蚀与膨胀间的对偶关系，我们可以将式（11.4.2）写成：

$$A \circledast B = (A \Theta B_1) - [A \oplus \hat{B}_2] \qquad (11.4.3)$$

然而，式（11.4.2）更为直观。上述三个公式定义了形态学上的击中与击不中变换。

11.5 几个基本的形态学算法

在前面的讨论的基础上，现在可以考虑一些形态学的实际用途。但处理二值图像时，形态学的主要应用是提取对于描绘和表达形状有用的图像成分。特别要考虑提取边界、连通分量和区域骨架的形态学算法。也要探讨击中变换与这些算法有联系的预处理或后处理方法（用预区域填充、细化和粗化）。

11.5.1 连接约定方式

在二值图像中，所谓的对象就是值为 1 且链接在一起的像素的集合。例如，以下所示

的矩阵就代表了包含一个简单的 3×3 方形对象的二值图像。矩阵中值为 0 的像素，表示图像的背景。

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

对于大多数操作来说，二值图像对象的识别依赖于确定图像中相邻像素时连接的约定方式。

MATLAB 通常使用两种连接方式：4－连接和 8－连接。

在 4－连接边沿约定方式中，所有与用户期望像素点接触的 8 个像素点中，水平垂直方向的 4 个像数均为可能连接的显示，如图 11-18。

在 8－连接边沿约定方式中，所有与用户期望像素点接触的 8 个像素点均为可能连接的显示，如图 11-19 所示。

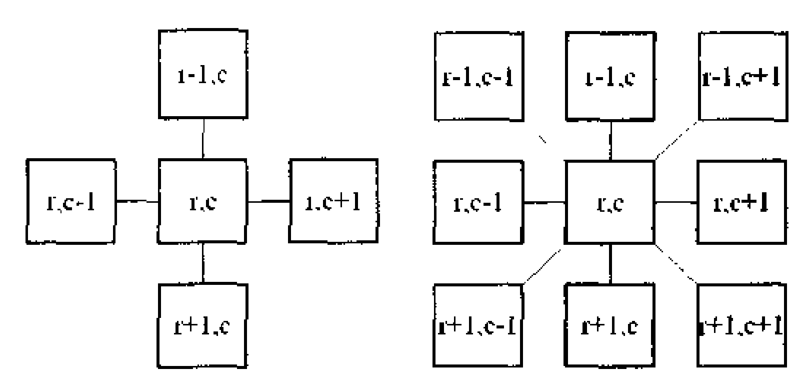


图 11-18　4-连接方式　　　图 11-19　8-连接方式

根据以上定义，下面的示例矩阵所表示的二值图像，在 4－连通边沿约定下包含三个对象，而在 8－连通边沿约定下包含两个对象。

示例矩阵：

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

4－连通边沿约定下的三个对象分别如下：

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 2 & 2 & 0 & 3 & 3 \\ 0 & 0 & 0 & 2 & 2 & 0 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 3 \end{bmatrix}$$

8－连通边沿约定下的两个对象分别如下：

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 1 & 1 & 0 & 2 & 2 \\ 0 & 0 & 0 & 1 & 1 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \end{bmatrix}$$

11.5.2 对象标注

MATLAB 的图像工具箱里面，提供了 bwlabel 函数作为二值图像连通区域的标注操作。在示例程序 11-7 定义了一个矩阵 I，然后示例程序 11-8 对其进行了对象标注，指定 4－连通边沿约定：

【示例程序 11-7】

```
I=zeros(8);
I(2:3,2:3)=1;
I(4:6,4:5)=1;
I(2:7,7:8)=1;
```

得到矩阵：

$$I=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

【示例程序 11-8】

```
%对上述 I
L=bwlabel(I,4)
```

得到结果为：

$$L=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 3 & 3 \\ 0 & 1 & 1 & 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 2 & 2 & 0 & 3 & 3 \\ 0 & 0 & 0 & 2 & 2 & 0 & 3 & 3 \\ 0 & 0 & 0 & 2 & 2 & 0 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

类似地，对相同进行了对象标注，指定 8－连通边沿约定：

【示例程序 11-9】

```
%对上述I
L=bwlabel(I,8)
```

得到的结果是：

$$L=\begin{bmatrix}0&0&0&0&0&0&0&0\\0&1&1&0&0&0&2&2\\0&1&1&0&0&0&2&2\\0&0&0&1&1&0&2&2\\0&0&0&1&1&0&2&2\\0&0&0&1&1&0&2&2\\0&0&0&0&0&0&2&2\\0&0&0&0&0&0&0&0\end{bmatrix}$$

11.5.3　边界提取

集合A的边界表示为$\beta(A)$，它可以通过先用B对A的腐蚀，而后用A减去腐蚀结果得到。即：

$$\beta(A)=A-(A\Theta B) \tag{11.5.1}$$

这里，B是一个合适的结构元素。

在MATLAB图像处理工具箱里面，提供了函数bwperin来提取二值图像中对象的边界像素。

示例程序11-10以blood1.tif的二值化图像进行边界识别，如图11-20所示。

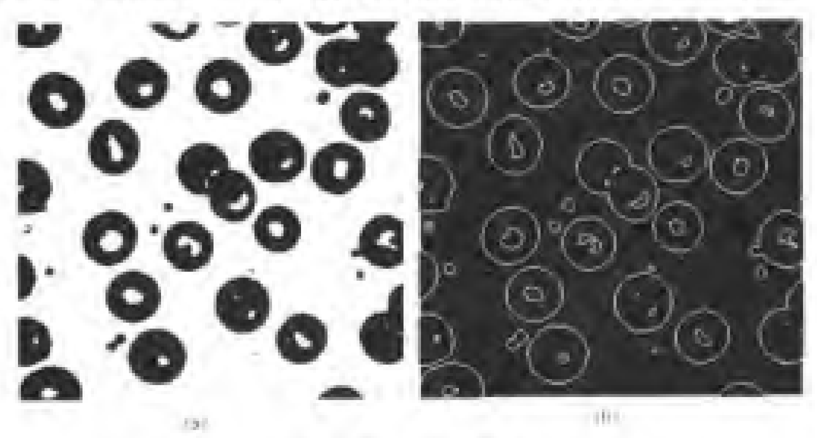


图11-20　边界提取

（a）二值图像。

（b）边界提取结果。

【示例程序11-10】

```
I=imread('blood1.tif');
J=im2bw(I);
figure,imshow(J);
BW1=bwperim(J);
figure,imshow(BW1);
```

对比结果，如上图11-20所示。

此外，还可以用前面提及的bwmorph函数（见示例程序11-11），得到同样的结果，如

图 11-21 所示。



图 11-21　用 bwmorph 函数提取边界

【示例程序 11-11】

```
I=imread('blood1.tif');
J=im2bw(I);
figure,imshow(J);
BW1=bwmorph(J,'remove');
figure, imshow(BW1);
```

11.5.4　区域填充

这是一个简单的用于区域填充的散发，它以集合的膨胀、求补和交集为基础。A 表示一个包含子集的集合，其子集的元素均是区域的 8－连通边界点，目的是从边界内的一个点开始，用 1 填充整个区域。过程为：

$$X_k=(X_{k-1}\oplus B)\bigcap A \quad k=1,2,3\ldots \tag{11.5.2}$$

其中，$X_0=p$，B 是一个适当的结构元素。如果 $X_k=X_{k-1}$，算法在第 k 步结束。X_k 和 A 的并集包含被填充的集合和它的边界。

MATLAB 图像处理工具箱提供了 bwfill 函数，可以进行区域填充。示例程序 11-12 对上述的 blood1.tif 进行求反后填充，得到的结果，如图 11-22 所示。

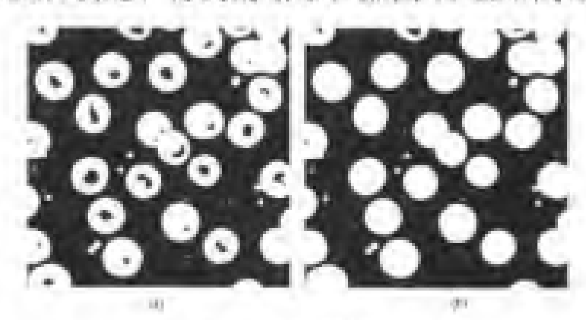


图 11-22　区域填充

（a）求反二值图。

（b）区域填充结果。

【示例程序 11-12】

```
I=imread('blood1.tif');
J=~im2bw(I);
figure,imshow(J);
BW1=bwfill(J,'holes');
figure,imshow(BW1);
```

11.5.5 细化

集合 A 使用结构元素进行细化，用 $A \otimes B$ 表示。细化过程可以根据击中或击不中变换定义：

$$A \otimes B = A - (A \circledast B) = A \cap (A \circledast B)^c \qquad (11.5.3)$$

这里仅对用结构元素进行模式匹配感兴趣，所以在击中或击不中变换中没有背景运算。相应的对于 A 的细化更为有用的一种表达方式是以结构元素序列为基础的：

$$\{B\} = \{B^1, B^2, B^3, \cdots, B^n\} \qquad (11.5.4)$$

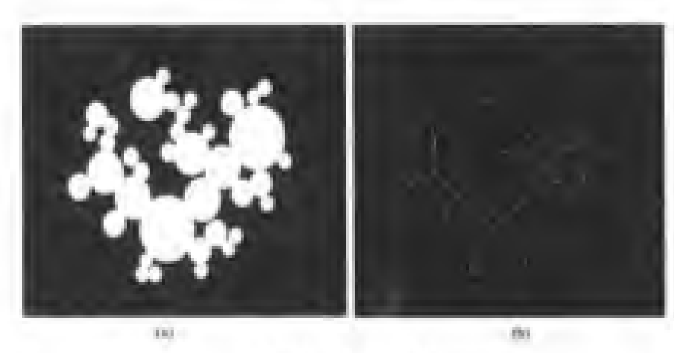
B^i

这里的 B^i 是 B^{i-1} 旋转后的形式。使用这个概念，现在用结构元素序列定义细化为：

$$A \otimes \{B\} = ((\cdots((A \otimes B^1) \otimes B^2) \cdots) \otimes B^n) \qquad (11.5.5)$$

这种处理通过使用 B^1 经过一遍处理对 A 进行细化，然后使用 B^2 经过一遍处理对 A 进行细化，如此进行下去，知道 A 使用 B^n 进行一次细化。整个过程不断重复，得到的结果不再发生变化。每遍独立的细化过程均使用式（11.5.3）执行。

细化操作依然可以用 MATLAB 里面的 bwmorph 函数实现。代码见示例程序 11-13，结果如图 11-23 所示。



(a) (b)

图 11-23 细化

（a）原图。

（b）细化结果。

【示例程序 11-13】

```
I=imread('circlesm.tif');
figure, imshow(I);
BW=bwmorph(I,'thin',Inf);
figure, imshow(BW);
```

11.5.6 粗化

粗化与细化在形态学上是对偶过程。它的定义如下：

$$A \odot B = A \cup (A \circledast B) \qquad (11.5.6)$$

这里 B 是适合与粗化处理的结构元素。和对细化的定义一样，粗化处理可以定义为一系列操作：

$$A \odot \{B\} = ((\cdots((A \odot B^1) \odot B^2) \cdots) \odot B^n) \qquad (11.5.7)$$

粗化的分离算法在实际使用中很少用到，代之而来的经常是相对所讨论的背景进行细化，而后对结果求补集。就是说，为了对集合 A 进行粗化，先令 $C = A^c$，而后对 C 进行

细化，最后再求补。

粗化操作同样可以用 MATLAB 里面的 bwmorph 函数实现。代码见示例程序 11-14，结果如图 11-24 所示。

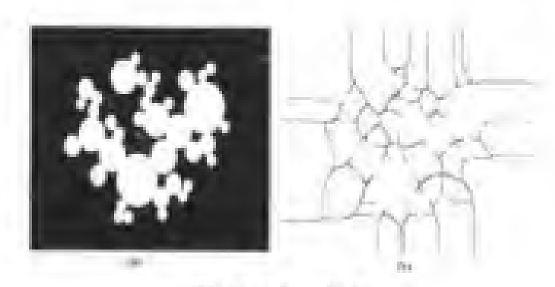


图 11-24 粗化

（a）原图。

（b）粗化结果。

【示例程序 11-14】

```
I=imread('circlesm.tif');
figure, imshow(I);
BW=bwmorph(I,'thicken',Inf);
figure, imshow(BW);
```

11.5.7 抽骨架

A 的骨架可以用腐蚀和开启得到。即骨架可以表示为：

$$S(A)=\bigcup_{k=0}^{n}S_k(A) \tag{11.5.8}$$

而：

$$S_k(A)=(A\Theta kB)-(A\Theta kB)\circ B \tag{11.5.9}$$

这里 B 是一个合适的结构元素，$(A\Theta kB)$表示对 A 的连续 k 次腐蚀：

$$(A\Theta kB)=(\cdots((A\Theta B)\Theta B)\Theta\cdots)\Theta B \tag{11.5.10}$$

第 k 次是 A 被腐蚀为空集前进行的最后一次叠代。这就是说：

$$K=\max\{k\mid A\Theta kB\neq\varnothing\} \tag{11.5.11}$$

式（11.5.8）和式（11.5.9）说明 S（A）可以由骨架子集$S_k(A)$的并集得到。即，A 可以通过使用下列公式由这些子集重构：

$$A=\bigcup_{k=0}^{n}(S_k(A)\oplus kB) \tag{11.5.12}$$

这里，$S_k(A)\oplus kB$表示$S_k(A)$的 k 次连续膨胀。

示例程序 11-15 对上述图片 circlesm.tif（图 11-24（a））的骨架提取，结果如图 11-25。



图 11-25 骨架提取

【示例程序 11-15】

```
I=imread('circlesm.tif');
figure, imshow(I);
BW=bwmorph(I,'skel',Inf);
figure, imshow(BW);
```

11.6 灰度图像扩展

这里简单地介绍一下灰度图像的基本形态学操作，包括：膨胀、腐蚀、开启与闭合。

用 b 对函数 f 进行的灰度膨胀表示定义为：

$$(f \oplus b)(s,t) = \max\{f(s-x,t-y)+b(x,y) \mid (s-x,t-y) \in D_f;(x,y) \in D_b\} \quad (11.6.1)$$

这里 D_f 和 D_b 分别为 f 和 b 的定义域。这里，f 与 b 都是函数，而不是二值形态学情况中的集合。

（s-x）与（t-y）必须在 f 的定义域内以及 x 和 y 必须在 b 的定义域内，条件与膨胀的二值定义中的条件是相似的（这里两个集合的交集必须至少有一个元素）。

灰度腐蚀，表示为 $(f\Theta b)$，定义为：

$$(f\Theta b)(s,t) = \min\{f(s-x,t-y)-b(x,y) \mid (s-x,t-y) \in D_f;(x,y) \in D_b\} \quad (11.6.2)$$

灰度图像的开启和闭合与二值图像的对应操作有相同形式。用结构元素 b 对图像 f 进行开启操作表示为 $f \circ b$，定义为：

$$f \circ b = (f\Theta b) \oplus b \quad (11.6.3)$$

同二值图像中的情况一样，开启先用 b 对 f 进行简单的腐蚀操作，而后用 b 对所得结果进行膨胀。

同样的用结构元素 b 对图像 f 进行开启操作表示为 $f \bullet b$，定义为：

$$f \bullet b = (f \oplus b)\Theta b \quad (11.6.4)$$

灰度图像的开启与闭合，对于求补和映射运算是对偶的。即：

$$(f \bullet b)^c = f^c \circ \hat{b} \quad (11.6.5)$$

小结

本章介绍了形态学的概念和技术，同时使用 MATLAB 作为实现工具，更形象的描绘了各种不同形态学运算的结果。本章所介绍的内容，不但是从图像中提取感兴趣特征的有力工具集，同时也将在后面的某些章节扮演重要角色。

重点：本章重点在于膨胀与腐蚀、开启与闭合、击中与击不中变换等一些基本的形态学算法。

难点：本章难点在于如何在实际的图像处理中应用形态学的算法达到处理效果。

疑点：形态学处理数字图像如何突破其局限性。

练习十一

一、选择题

1. 定义 A（x，y）为异或运算，则 A（128，128）=（　）。

A. 0　　B. 1　　C. 2　　D. −1

2. 开运算是指（ ）的过程。

A. 腐蚀 B. 膨胀 C. 先膨胀后腐蚀 D. 先腐蚀后膨胀

二、思考题

1. 用表达式写出下列图形中的阴影部分，集合 A、B、C 的位置如图 11-26 所示。

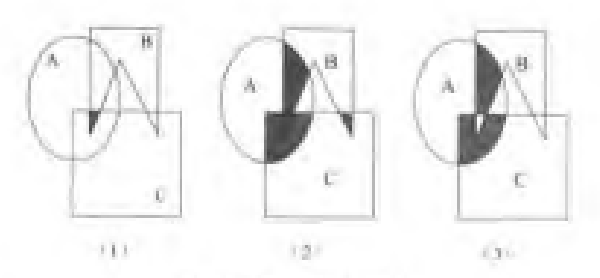


图 11-26 集合运算

2. 写出下列矩阵按 4－连接和 8－连接两种不同连接方式标记连通区域的结果：

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

三、上机题

读入二值图 circlesm.tif，对其添加椒盐噪声，结合本章知识，把噪声滤除。

第 12 章　图像压缩技术简介

图像处理的应用主要有两个方面，一个是模式识别，另一个就是图像压缩技术。在图像数字化中，图像的原始数据是十分巨大的，图像压缩的目的是节省图像存储器的容量，减少传输信道容量，缩短图像处理的时间等等，以使图像能在实际中更广泛地得到应用。

本章的主要内容如下：

（1）绪论。

（2）数据压缩算法。

（3）无损压缩技术。

（4）有损压缩技术。

12.1　绪论

1948 年，Oliver 提出 PCM（脉冲码调制编码）。1952 年，Huffman 提出根据字符出现的概率来构造平均编码长度最短的异字头码字，有时称为最佳编码。1977 年，Lempel 和 Ziv 提出基于字典的编码。80 年代初，第一代压缩编码已成熟，进入实用阶段。80 年代中期，开始了第二代压缩编码的研究。1989 年，制成数据压缩集成电路。

12.1.1　数据的压缩

图像数据文件通常包含着数量可观的冗余信息，另外还有大量的不相干的信息，这就为现代数据压缩技术提供了可能。

数据压缩技术利用了数据固有的冗余性和不相干性，将一个大的数据文件转换成较小的文件。由压缩的文件在以后需要的时候以精确的或近似的方式将原文件恢复出来。两个文件的大小之比（压缩比）确定了压缩的程度。

信源编码：主要解决有效性问题。通过压缩、扰乱、加密处理，力求用最少的数码传递最大的信息量，使信号更适宜传输。

信道编码：主要解决可靠性问题。即尽量使处理过的信号在传输过程中不出错，并能自动检错和纠错。

信号空间：物理空间、时间空间和电磁频谱空间。

由此得出数据压缩的定义：以最少的数码表示信源所发的信号，减少容纳给定消息集合或数据采样集合的信号空间。

12.1.2　数据压缩的必要性

1.　数字信号的传输速率或比特率

$I = f_s \times R$　　（bit/s 或 b/s）　　（12.1.1）

其中 f_s 为取样频率，R 为每个取样的幅度值用二进制编码的位数。

2.　实例

数字电话取样率为 $8KHz$，每取样用 8 位编码，数码率为：

$I = 8 \times 8 = 64kb/s$。

一台彩色数字电视，用12.5/6.75/6.75 MHz频率采样，每像素用8位编码，数码率为：

$I = (13.5 + 6.75 + 6.75) \times 8 = 216Mb/s$

若实时传送，需占上述数字话路3375个。若压缩到原来1/3，可同时增开2250路数字电话。由此可见，数据压缩在应用中是十分必要的。

12.1.3 压缩的条件

数据压缩是有条件的，主要表现在以下两个方面：

（1）数据冗余度。

音频信号和视频信号等原始数据通常存在很多用处不大的空间，空间越多，数据的“冗余度”也越大。通过数据的压缩，将把这些不用的空间去掉。

（2）人类不敏感因素。

一般而言，人类对某些频率的音频信号不敏感，有无这些频率的音频，在听觉上影响不大，在数据压缩时，就可去掉这些不敏感的成分，以使减少数据量。

根据人眼对物体的轮廓比对物体内部细节更为敏感的特点，可以利用物体（而不是像素）的集合来表达图像。

人眼对彩色细节的分辨能力远比亮度细节的分辨能力低。

12.1.4 数据冗余

1. 冗余的基本概念

冗余是指信息所具有的各种性质中多余的无用空间，其多余的程度叫做“冗余度”。

信息量、数据量和冗余量之间的关系：I = D-du

式中，I代表信息量，D表示数据量，du是冗余量。

例如180个汉字，可以用两种方法存储：

（1）180个汉字 = 360字节，需0.36KB。

（2）用语音存放，若读180个汉字需一分钟，采样频率为8KHz，采样精度为8bit，则存一分钟的音频信号，需：

(8KHz × 8bit/8) × 60s = 480KB

两者相差1000倍。

又例如：数据与信息，冗余信息和不相关的信息

你的妻子王芳将于3月11日星期一晚上10点5分到达广州白云机场，请接机。

无有损压缩：王芳将于3月11日22点5分到达白云机场，请接机。

有损压缩：芳将于3月11日晚上10点到达白云机场。

2. 冗余分类

（1）空间冗余——规则物体的表面具有物理相关性，将其表面数字化后表现为数据冗余。

（2）时间冗余——视频信号和动画等有序排列的图像很容易产生数据冗余现象。在播放有序排列图像时，相邻画面无变化的内容构成了时间上的冗余。

（3）统计冗余——是空间冗余和时间冗余的总称。在数据处理时，往往采用统计出

现概率的办法来鉴别空间冗余和时间冗余，因此空间冗余和时间冗余具有统计特性。

例如，某图像中，相邻的相同特性像素重复出现的概率非常大，则相邻像素具有相关性，即被确认有冗余发生。而图像的其他像素重复出现概率很小，相邻像素的相关性不大，冗余就会很小，或不发生冗余。

（4）结构冗余——在数字化图像中，具有规则纹理的表面，大面积相互重叠的相同图案、规则有序排列的图形等结构，都存在数据冗余，这种结构上的冗余叫做“结构冗余”。在处理图像时，只须把基本形状的像素进行存储或传输，其余部分利用坐标描述即可。

（5）信息熵冗余——信息熵冗余也叫“编码冗余”。信息熵是指一团数据所携带的单位信息量。

信息熵的一般定义为：

$$H=-\sum_{j=0}^{k-1}p_j\bullet\log p_j \qquad (12.1.2)$$

式中，H 为信息熵，k 为数据类数或码元的个数，p_j 为发生概率。

单位数据量 d 定义为：

$$d=\sum_{j=0}^{k-1}p_j\bullet b(y_j) \qquad (12.1.3)$$

式中，d 为单位数据量，K 为数据类数或码元的个数，$b(y_j)$为分配给码元类 y_j 的比特数，通常 d>H。

（6）视觉冗余——人类的视觉敏感度一般小于图像的表现力，图像的微小色彩变化、亮度层次的细腻变化以及轮廓的细微差别不易察觉，这就产生了视觉冗余。

（7）知识冗余——知识是人类独有的，是认知自然、总结规律而得到的。人类一旦掌握了知识，凭借经验就可辨识事物，无须进行全面的比较和鉴别，而计算机在处理数据时，没有经验可循，只是按部就班地处理数据。这种与人类差异造成的数据冗余就叫做“知识冗余”。

12.1.5 数据压缩的标准和应用

国际标准化组织（ISO）、国际电工委员会（IEC）和国际电信联盟的电信标准部（ITU - T 或 ITU - TS）陆续制订的各种数据压缩与通信的标准和建议。例如由国际标准化组织（ISO）和 CCITT 联合发起的联合图像专家组（JEPG），建立了静态图像压缩的公开算法。在视觉效果不受到严重损失的前提下，算法可以达到 15 或者 20 的压缩比。如果在图像质量上稍微牺牲一点的话，可以达到 40：1 或更高的压缩比。

如果处理的是彩色图像，JEPG 算法首先将 RGB 分量转化成亮度分量和色差分量，同时丢失一半的彩色信息（空间分辨率减半）。然后用 DCT 来进行块变换编码，舍弃高频的系数，并对余下的系数进行量化以进一步减小数据量。最后使用 RLE 和 Huffman 编码来完成压缩任务。JEPG 的解压缩过程就是 JEPG 压缩过程的逆过程，这使得算法具有对称性。

由运动图像专家组（MPEG）提出的另一个公开算法，用于压缩附加声音的全运动图像。它与 JEPG 算法在概念上类似，只不过它还利用了相邻图像之间的冗余信息。

由于可达到 100:1 的压缩比，所以 MEPG 算法非常实用，如用于在每秒一兆位的信道中传送带声音的彩色电视图像，以及在磁盘驱动器中储存较长一段时间的数字电视图像片

段等。

为 JEPG 和 MEPG 压缩与解压缩所专门设计的高速硬件，可以极大地降低计算量上的负担。

12.2 数据压缩算法

数据压缩的核心是计算方法，不同的计算方法，产生不同形式的压缩编码，以解决不同数据的存储与传送问题。

数据冗余类型和数据压缩的算法是对应的，一般根据不同的冗余类型采用不同的编码形式，随后是采用特定的技术手段和软硬件，以实现数据压缩。

数据压缩算法分类：

（1）无损压缩编码：解码后的数据与压缩之前的原始数据完全一致。

（2）有损压缩编码：解码后的数据与原始数据不一致。

1. 无损压缩编码

无损压缩编码基于信息熵原理，属于可逆编码。其压缩比一般不高。

所谓“可逆”，是指压缩的数据可以完全地还原成原始数据。

典型的可逆编码有：霍夫曼编码、算术编码、行程编码等。

2. 有损压缩编码

该编码在压缩时舍弃部分数据，还原后的数据与原始数据存在差异。有损压缩具有不可恢复性和不可逆性。

有损压缩编码类型有：

（1）预测编码——基于线性预测原理的编码，主要用于对数据冗余进行压缩。由于图像中相邻像点的相关性较强，若其中一点已经被编码，便可预测并估计相邻像点的编码模式。

（2）PCM 编码——脉冲编码调制编码，主要用于对连续语音信号进行空间采样、量化和进行数字编码。只要采样频率足够高，采样精度足够大，解码后的数字音频信号质量也就比较高。这种对声音直接量化的方法数据量大，要求信号传输速率高。

（3）量化与向量量化编码——基于向量量化原理的编码。把模拟量转换成数字量需经过量化过程，若量化数据在动态范围内的概率密度呈均匀分布的话，则可等间隔地区分量化的级别。对图像的像点进行量化时，一般可每次量化一个像点，但也可量化一组像点。量化一组像点的做法叫做“向量量化”。

（4）频段划分编码——基于频段划分处理原理的编码。当图像数据变换到频域后，按照频率分布划分频段，然后对各频段进行不同的量化处理，使组合方式达到最优。

（5）变换编码——基于正交变换原理的编码，这种编码主要用于对统计冗余和视觉冗余进行压缩。如 K-L 变换和小波变换。

（6）知识编码——基于知识的编码。将人类知识用参数进行描述，形成一个规则库。然后再根据规则库中的参数，对图像进行编码和解码。

12.3 无损压缩技术

无损压缩技术主要有基于字典的技术、统计编码和其他编码。

12.3.1 基于字典的技术（定长编码）

1. 游程编码（RLE）

图像灰度级较少。

具有相同灰度值的一些像素组成的序列称为一个游程。

每个游程用下列一个记录存放：

像素值	标志性字符	游程长度

最坏时，文件长度加倍。

2. LZW 编码（文本编码）

由 Lemple 和 Ziv 首先提出无损压缩技术的 LZ 编码，它由 Welch 加以充实而形成 LZW。

基于字典，将输入字符串映射成定长（通常为 12 位）的码字。

在 12 位 4096 种可能的代码中，256 个代表单字符，剩下 3840 个代表出现的字符串。

LZW 字典中的字符串具有前缀性，即 $\omega K \in T \Rightarrow \omega \in T$ 。

LZW 码能有效利用字符出现频率冗余度进行压缩，且字典是自适应生成的。

LZW 算法流程：

初始化：将所有的单字符串放入串表

读第一个输入字符 ——→ 前缀串 ω

Step: 读下一个输入字符 K

if 没有这样的 K（输入已穷尽）：

码字(ω)——→ 输出；结束。

if ωK 已存在于串表中：

ωK——→ω；repeat Step.

else ωK 不在于串表中：

码字(ω)——→ 输出

ωK——→ 串表

K——→ω；repeat Step.

12.3.2 统计编码方法（不定长编码）

1. 消息的自信息量与信息熵

假定无记忆信息源，可发出 K 种信息。字母表为：

$$A_K = \{a_j\}, j = 0,1,...K-1 \qquad (12.3.1)$$

每个符号出现的概率为：$p_j = p(a_j)$,

$$\sum_{j=0}^{K-1} p(a_j) = 1 \qquad (12.3.2)$$

则字符 a_j 出现的自信息量定义为：

$$I(a_j) = -\log_r p_j \qquad (12.3.3)$$

当 $r = 2$ 时，单位为比特（bit）。

当 $r = e$ 时，单位为奈特（Nat）。

当 $r=10$ 时，单位为哈特（Hart）。

$I(a_j)$ 称为自信息函数，含义为：随机变量 X 取值为 a_j 时所携带信息的度量。

自信息函数表述，越不可能出现的符号，它的出现对于消息的信息量的贡献越大。

将自信息量的概率平均值，即随机变量 $I(a_j)$ 的数学期望值称为信息熵，定义为：

$$H(X)=H(p_1,p_2...,p_k)=\sum_{j=0}^{K-1}p_j\bullet I(a_j)=-\sum_{j=0}^{K-1}p_j\bullet\log p_j \tag{12.3.4}$$

单位为 bit/字符。

它表示字母表 A_K 中每个字符出现的平均不确定性，或每出现一个字符所给出的平均信息量，或为记录 A_K 中每个字符的出现所需的平均编码长度。

定理 1（最大离散熵）：所有概率分布 p_j 所构成的熵，以等概率时为最大，即：

$$H(X)=H(p_1,p_2...,p_k)\leqslant\log K \tag{12.3.5}$$

定理 2：

$$H(X)=-\sum_{j=0}^{K-1}p_j\bullet\log p_j\leqslant-\sum_{j=0}^{K-1}p_j\bullet\log q_j \tag{12.3.6}$$

其中等号仅在 $\{q_j\}=\{p_j\}$ 时成立。

H(X)给出了无损压缩的极限。

H(X)给出了熵编码的方法：对于离散无记忆的信源，必须：

（1）准确的到字符概率 $\{p_j\}$。

（2）每个字符的编码长度都达到它的自信息量。

设有一种编码方法，其字母 a_j 的码字长度为 $L_w(a_j)$，显然 $L_w(a_j)$ 也是一个非负的随机变量，记：

$$L_w(a_j)=-\log q_j\quad (0<=q_j<=1, j=1,2,\ldots,k,\ \sum_{j=0}^{K-1}q(a_j)=1) \tag{12.3.7}$$

则该编码的冗余量为：

$$R=E\{L_w(a_j)\}-H \tag{12.3.8}$$

当 $L_w(a_j)=-\log[p(a_j)]$ 时，称该编码为最佳编码。

2. Huffman 编码

根据字符出现的概率来构造平均长度最短的异字头码字。它使得编码的冗余量最小。

定理 3：在变长编码中，若各码字长度严格按照所对应符号出现概率的大小逆序排列，则其平均长度为最小。

实现上述定理的编码步骤如下：

（1）将信源符号出现概率按降序排列。

（2）将两个最小的概率进行组合相加，并继续这一步骤，始终将较高的概率分支放在上部，直到概率达到 1.0 为止。

（3）对每对组合中的上边一个指定为 1，下边一个指定为 0（或相反：对上边一个指定为 0，下边一个指定为 1）。

（4）画出由每个信源符号概率到 1.0 处的路径，记下沿路径的 1 和 0。

（5）对于每个信源符号都写出 1、0 序列，则从右到左就得到霍夫曼码。

12.4 有损压缩技术

12.4.1 标量量化（对应矢量量化）

1. 均匀量化

设 $x\in[a_l,a_u]$ 为量化器输入信号幅值，$p(x)$ 为其概率密度函数，则有：

$$\int_{a_L}^{a_u} p(x)dx = 1 \tag{12.4.1}$$

记量化总层数为 J，$d_k(k=0,1,...J)$ 为判别电平，当 $d_k < x \le d_k+1$ 时，输出量化值 y_k，量化误差为 $x-y_k$。

最简单的量化方法（均匀量化）：

$$\begin{cases} d_{k+1}-d_k = d_k - d_{k-1} = \Delta \\ \quad y_k = \dfrac{d_{k-1}+d_k}{2} \end{cases}, \quad k=1,2,...J-1 \tag{12.4.2}$$

2. 最佳量化

按均方误差最小化来定义最佳量化，即：

$$\varepsilon = E\left\{(x-y)^2\right\} = \int_{a_l}^{a_u}(x-y)^2 p(x)dx = \sum_{k=0}^{J-1}\int_{d_k}^{d_{k+1}}(x-y_k)^2 p(x)dx \tag{12.4.3}$$

求极值，可得：

$$d_k = \frac{y_{k-1}+y_k}{2}, \quad k=1,2,...,J-1 \tag{12.4.4}$$

及：

$$y_k = \frac{\int_{d_k}^{d_{k+1}} xp(x)dx}{\int_{d_k}^{d_{k+1}} p(x)dx} \tag{12.4.5}$$

由 $M-L$ 迭代算法求出。

12.4.2 率失真函数理论

率失真理论寻求定字长的编码策略的失真度与码率之间的关系。

问题：在给定的失真条件下，最起码需要多大的码率，才能保证不超过允许的失真。

比如信源符号有八种：

$$\begin{Bmatrix} a_i: & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ b_i: & 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \end{Bmatrix}$$

那么只要传输不错，则收到一码字。例如 8 对应 111。即平均失真为 0，码率为 3。

若允许一定误差，则码率可缩短，例如：

$$\begin{Bmatrix} 1,2 & 3,4 & 5,6 & 7,8 \\ 00 & 01 & 10 & 11 \end{Bmatrix}$$

显然允许失真度 D 越大，码率 $R(D)$ 越小。

当使用有损压缩时，重构图像 $g(x,y)$ 将与原始图像 $f(x,y)$ 不同。二者的差别（失真度）可以很方便地由重构的均方误差来定量确定：

$$D = E\left\{[f(x,y) - g(x,y)]^2\right\} \tag{12.4.6}$$

其中 $f(x,y)$ 原图，$g(x,y)$ 为压缩后的重建图像。如果我们定义一个最大容许失真量 D^*，那么编码时对应的比特率的下限 $R(D^*)$ 成为率失真函数。

重构误差的熵由下式得出：

$$H[f(x,y) - g(x,y)] \leqslant \frac{1}{2}\log(2\pi e D^*) \tag{12.4.7}$$

定理 4（信源编码定理）：一个具有率失真函数 $R(D)$ 的信源，若允许失真范围为 D，并有两个任意小的正数 ε 与 δ，则必存在一种信源编码、译码方法，使码率 $R > R(D)+\delta$，而平均失真 $\overline{D} \leqslant D+\varepsilon$。

小结

图像压缩降低了数字图像对存储量的要求，缩短了图像传送所需要的时间，但是这是以压缩和解压缩的时间为代价的。有损压缩比无损压缩可以达到更高的压缩比，但是后者保持了数据的完整性。率失真理论提供了用来判定在什么条件下固定码长的图像编码的性能可以达到最优。并且率失真函数给出了固定码长的编码在指定的最大失真率水平下信率的下界。图像压缩技术在现实生活中有着非常广泛的应用，随着应用领域的推广，压缩技术也将飞速发展。

重点：本章重点在于了解图像压缩在数字图像处理中的重要性，以及常用的压缩算法。

难点：如何运用各种手段（例如小波）进行图像压缩。

疑点：在选择一种压缩算法时，无损压缩还是有损压缩如何选择。

练习十二

一、选择题

1. ISO 和 CCITT 于 1986 年成立的联合图像专家组，于 1992 年推出的静止图像压缩算法是（　）。

A. H.261　　B. JEPG

C. MPEG-1、MPEG-2　　D. H.263

2. 提出根据字符出现的概率来构造平均编码长度最短的异字头码字的编码叫（　）。

A. 霍夫曼编码　　B. 字典编码　　C. 脉冲码调制编码　　D. 子带编码

二、思考题

1. 数据压缩的条件是什么？

2. 数据冗余主要分为几大类？

三、上机题

用基于 DCT 变换的图像压缩技术压缩图像，对比压缩比分别为 4:1 和 8:1 时图像的效果。

第 13 章　彩色图像处理

对彩色图像进行处理主要源于以下两个因素：

（1）颜色是一个强有力的描绘子，它不仅可以使目标物的区分变得简单，而且简化了从场景中抽取目标的过程。

（2）人可以辨别几千种颜色色调和亮度，相形之下却只能辨别几十种灰度层次。

其中，第二个因素对于人工图像分析特别重要。

彩色图像处理主要分为两个领域：全彩色处理和伪彩色处理。在第一类中，图像用全彩色传感器获取，如：彩色电视摄像机或彩色扫描仪。在第二类中，对特定的单一亮度或亮度范围赋予--种颜色。到目前为止，彩色图像处理多为伪彩色处理。然而，在过去的十年，处理彩色图像的彩色传感器和硬件的价格变得更容易接受，使得全彩色图像处理技术的应用日益广泛，包括印刷、可视化和互联网应用等。显然，前面各章提到的对灰度图像处理的方法可直接用于彩色图像处理，其余的将按照本章介绍的彩色空间特性加以重新阐述。

本章主要内容如下：

（1）人眼视觉原理及彩色基础。

（2）彩色模型及彩色坐标转换。

（3）彩色变换。

（4）彩色图像的平滑和锐化。

（5）彩色图像分割及边缘检测。

（6）伪彩色和假彩色处理。

（7）彩色图像的噪声。

（8）彩色图像的压缩。

13.1　人眼视觉原理及彩色基础

1666 年，Isaac.Newton 发现，当一束太阳光通过一个玻璃棱镜的时候，会出现一端为紫色一端为红色的连续彩色谱，且该彩色谱可分为六个区域：紫色、蓝色、绿色、黄色、橘红色、红色，彩色谱中相邻颜色间平滑过渡。

人类和某些动物接收到一个物体的颜色是由物体反射光的性质决定的。可见光由电磁波谱中相对较窄的波段组成，一个物体反射的光如果在所有可见光波长范围内是平衡的，对观察者来说显示白色；若一个物体对有限的可见光波长范围反射，则物体呈现某种颜色。例如：绿色物体反射具有 500nm ~ 570nm 范围的光，吸收其他波长光的多数能量。

人的视网膜由感光细胞覆盖，感光细胞吸收来自光学图像的光线，并通过晶体透镜和角膜聚焦在视网膜上生成神经脉冲，通过大约一百万个光学神经纤维传送到大脑，这些脉冲的频率代表了入射光线的强度。视网膜中的感光细胞根据其物理形状分为杆状细胞和锥状细胞。杆状细胞只能感光，且感光能力较为敏感，对弱光的灵敏度要远远超过锥状细胞，但对正常光照却失去作用；锥状细胞不但能感光，而且能感色，但锥状细胞只有在正常光

照的作用下才能引起视觉并产生色感。

按锥状细胞将光信号转化为神经脉冲的感光特性，将其区分为红、绿、蓝三类锥状细胞，红、绿、蓝三种颜色被称为人类视觉的三基色。图13-1和图13-2表示了人眼（右眼）和人类视觉系统中三类锥状细胞的光谱敏感曲线。当一束光线射入肉眼时，三种锥状细胞就产生不同的反应，例如，当一束青光射入视网膜时，只对绿、蓝锥状细胞产生刺激，引起青色视觉；当一束黄光射入视网膜时只对红，绿锥状细胞产生刺激，引起黄色视觉等等。不同颜色的光对三种细胞的刺激量不同产生了五光十色的颜色，肉眼感觉到的各种颜色就是不同波长的光刺激三基色细胞的结果。

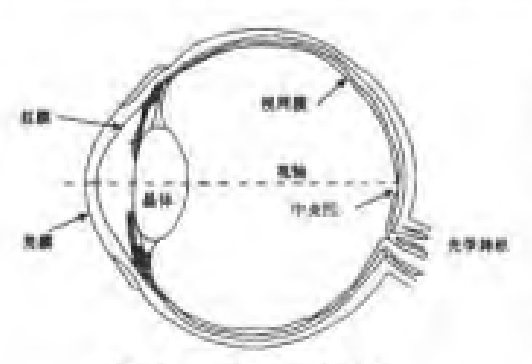



图13-1 人眼（右眼）

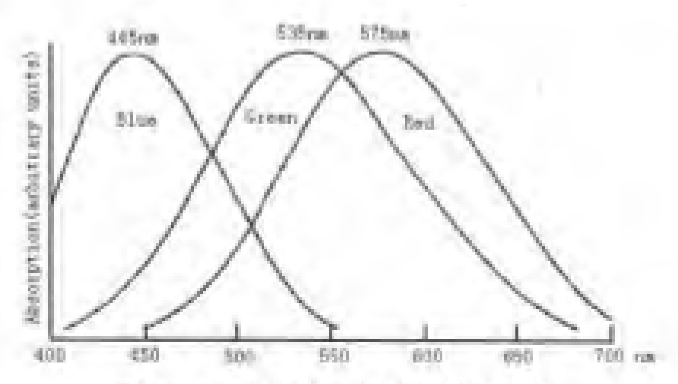


图13-2 人类感光细胞的敏感曲线

感光细胞经过一层双极性细胞和一层神经节细胞后生成神经脉冲，人工神经元网络就是以这些视网膜细胞的结构和操作为模型的。一百万左右的神经节细胞的轴突纤维形成了光神经，它们将图像数据传送到大脑。

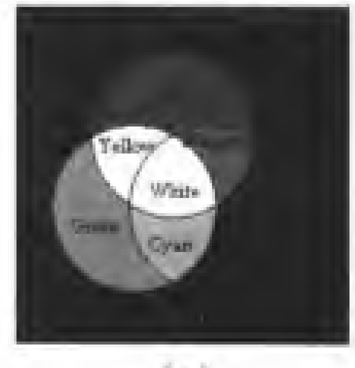
原色相加可产生二次色，如：

品红色（Magenta）= 红（Red）+ 蓝（Blue）。

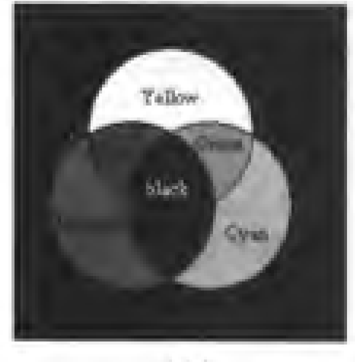
青色（Cyan）= 绿（Green）+ 蓝（Blue）。

黄色（Yellow）= 红（Red）+ 绿（Green）。

如果以合适的亮度把三原色或者一种二次色与其相反的原色相混合，就可以得到白光。这一结果如图13-3（a）所示，它还说明三原色两两相混合产生二次色。



（a）



（b）

图13-3 光及颜料的原色和二次色

光原色与颜料或着色剂原色之间有所区别，后者定义一种原色为减去或吸收光的一种原色，并反射或传输另两种原色。因此，颜料的原色是品红色（Magenta）、青色（Cyan）和黄色（Yellow），而二次色是红（Red）、绿（Green）和蓝（Blue）。如：

青色（Cyan）= 白色（White）- 红色（Red）。

品红色（Magenta）= 白色（White）- 绿色（Green）。

黄色（Yellow）= 白色（White）- 蓝色（Blue）。

蓝色 = 白色（White）- 红色（Red）- 绿色（Green）。

绿色 = 白色（White）- 红色（Red）- 蓝色（Blue）。

红色 = 白色（White）- 绿色（Green）- 蓝色（Blue）。

黑色 = 白色（White）- 蓝色（Blue）- 红色（Red）- 绿色（Green）。

如图 13-3（b）给出了这些颜色，这三种颜料原色或者其相对应的补色进行合适的组合产生黑色。彩色电视接收机便是通过色光相加产生彩色的一个实例：彩色电视显像管内部是由电敏荧光粉三角形点阵形式组成的，当激发时，三色组中的每一点能产生三原色中的一种光，发射红光的荧光粉的像点亮度由显像管内的电子枪调制，该电子枪产生的脉冲与电视摄像机摄取的"红能量"相对应。三像素组中的绿点和蓝点荧光物以相同的方式被调制，在电视接收机上观察到的效果是，三像素组每一荧光点原色加在一起并由眼睛对颜色敏感的锥状体以全彩色图像的方式接收，以这三种颜色方式每秒钟连续改变 30 幅画面，在荧光屏上可完成连续的彩色图像的显示。

13.2 彩色模型和彩色坐标转换

彩色模型（也称彩色空间或彩色系统）的用途是在某些标准下，用通常可接受的方式简化彩色规范。本质上，彩色模型是坐标系统和子空间的规范，系统中的每种颜色都由单个点表示。

现在所用的大多数彩色模型都是面向硬件（如：彩色监视器和打印机）或者面向应用的。在数字图像处理中，最通用的面向硬件的模型是 RGB（红、绿、蓝）模型。该模型用于彩色监视器和很多种彩色视频摄像机；CMY（青、品红、黄）、CMYK（青、品红、黄、黑）模型是针对彩色打印机的；HSI（色调、饱和度、亮度）模型更符合人们描述和解释颜色的方式，而且，HSI 模型还有一个优点，就是把图像分成彩色和灰度信息，使其更适合许多灰度处理技术。由于彩色科学是一个包括很多应用场合的非常宽的领域，因此现在使用的彩色模型还有很多。本章将较为详细的介绍在图像处理领域几个重要的彩色模型。

目前广泛采用的彩色信息表达方法都是在三基色按比例混合的配色方程基础上得到的。配色方程为：

$$C = aR + bG + cB \tag{13.2.1}$$

其中 C 为任意一种彩色光，a、b、c 为三基色 R、G、B 的权值。配色方程与肉眼视网膜中的三种色敏锥状晶体是一致的。这三种锥状晶体对可见光谱中黄 - 绿区、绿区与蓝区分别呈现尖峰反应。根据式（13.2.1），一幅彩色图像中的每一个像素都可以用三维色空间（R，G，B）中的一个矢量$[a, b, c]^T$来表示。R、G、B 的比例关系确定了所配彩色光的色度（包含色调和饱和度），其数值确定了所配彩色光的光通量，aR、bG、cB 分别代表彩色 C 中所含三基色的光通量成分，又称为彩色分量。

国际照明委员会（CIE）通过配色实验确定了基色光波长，它们分别为 $\lambda_R = 700nm$，$\lambda_G = 546.1nm$，$\lambda_B = 435.8nm$，并按 1.0000:4.5907:0.0601 三基色光通量的比例得到 5.6508nm

的白光的能量。由于该三基色是由 CIE 规定的，所以标记为 R_{CIE}，G_{CIE}，B_{CIE}。这种配色过程的物理意义很明确，但定量计算时比较复杂，而且，在某些情况下颜色还会出现负值。为了克服这些缺点，CIE 又提出了 XYZ 计色制，在 XYZ 计色体系中，X、Y、Z 并不代表真实的物理色彩。

13.2.1 XYZ 计色体系

在 XYZ 计色体系中，令 X、Y、Z 三个基色代表 R、G、B 三基色，建立新的计算系统和色度图。X、Y、Z 与 R、G、B 基色体系之间有如下线性关系：

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.490 & 0.310 & 0.200 \\ 0.177 & 0.813 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{bmatrix} \begin{bmatrix} R_{CIE} \\ G_{CIE} \\ B_{CIE} \end{bmatrix} \tag{13.2.2}$$

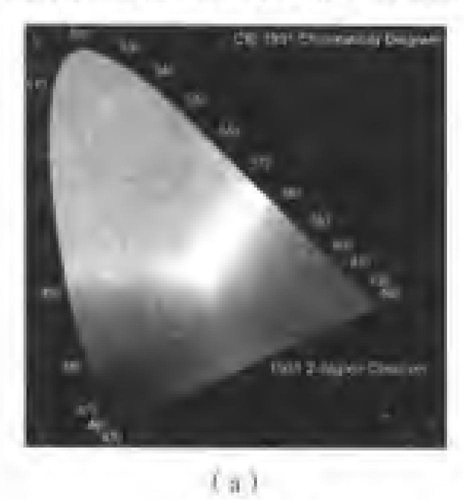
标准白光可由 X = Y = Z = 1 来表示。色度坐标为：

$$x = \frac{X}{X+Y+Z} = \frac{X}{m} \tag{13.2.3}$$

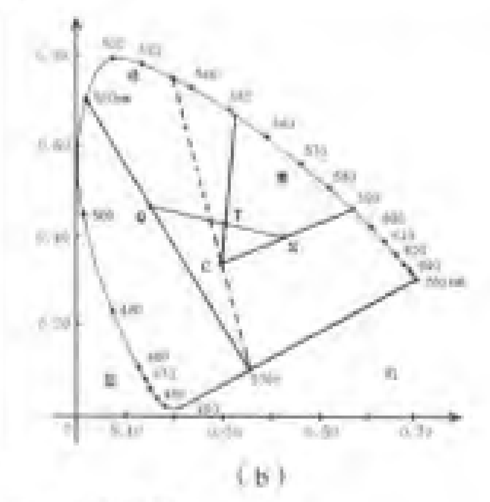
$$y = \frac{Y}{X+Y+Z} = \frac{Y}{m} \tag{13.2.4}$$

$$y = \frac{Z}{X+Y+Z} = \frac{Z}{m} \tag{13.2.5}$$

其中，$m = X + Y + Z$,m 为色模，表示某彩色光所含标准三基色系数的总量；色度是由三色系数 X、Y、Z 的相对值确定的，与 X、Y、Z 的绝对值无关；X、Y、Z 为相对色度系数，又叫做色度坐标。由式（13.2.3）、（13.2.4）、（13.2.5）可知，$x+y+z=1$。由此可得，当 x、y 已知时，则 z 也为已知，z 是一个非独立参量。这样可将配色实验中得到的三基色转换到 x、y 坐标系中，得到图 13-4（a）所示的色度图。在色度图中，从 380nm 的紫色到 780nm 的红色的各种谱色的位置标在舌形色度图周围的边界上，任何不在边界上而在色度图内部的点都表示谱色的混合色。靠近图中心的 C 是白色，相当于中午阳光的光色，其色度坐标为 x = 0.3101，y = 0.3162。图 13-4（b）下方的直线部分，即连接 400nm 和 700nm 的直线，由紫到红的系列，是光谱上所没有的。



（a）



（b）

图 13-4 CIE 1931 色度图

假设色度图上有一颜色 S，由 C 通过 S 画一直线至光谱轨迹 0 点（590nm），则 S 颜色

的主波长即为 590nm，此处光谱的颜色即 S 的色调（橙色）。某一颜色离开 C 点至光谱轨迹的距离表明它的色纯度，即饱和度。颜色越靠近 C 越不纯，越靠近光谱轨迹越纯。S 点位于从 C 到 590nm 光谱轨迹的 45%处，所以它的色纯度为 45%（色纯度% =（CS / CO）×100）。从光谱轨迹的任一点通过 C 画一直线抵达对侧光谱轨迹的一点，这条直线两端的颜色互为补色（虚线）。从紫红色段的任一点通过 C 点画一直线抵达对侧光谱轨迹的一点，这个非光谱色就用该光谱颜色的补色来表示。表示方法是在非光谱色的补色的波长后面加字母“C”，如 536C，表示这一紫红色是 536nm 绿色的补色。

13.2.2 UCS 均匀色标体系

在均匀色标 UCS（Uniform Chromatic Scale）体系中，坐标为 Y、U、V，其中 Y 为亮度，U、V 为色度，U、V 与色度坐标 X、Y 表示的色度之间的关系式为：

$$U=\frac{4X}{X+15Y+3Z}=\frac{4x}{-2x+12y+3} \tag{13.2.6}$$

$$V=\frac{6Y}{X+15Y+3Z}=\frac{6y}{-2x+12y+3} \tag{13.2.7}$$

在 UCS 体系中，相同长度的线段所代表的色度差别与人的视觉结果是一致的，而且在任何方向上基本相同。1964 年，CIE 对 UCS 体系又作了改进，推出了一种新的均匀颜色空间体系 ULCS（Uniform Lightness and Chromatic Scale）。ULCS 体系是通过 UCS 坐标系平移得到的，其色度坐标 U^*、V^*、W^* 与 UCS 中的 U、V 间的关系为：

$$W^*=25(100Y)^{\frac{1}{3}}-17 \qquad 0.01\leqslant Y\leqslant 1 \tag{13.2.8}$$

$$U^*=13W^*(U-U_0) \tag{13.2.9}$$

$$V^*=13W^*(V-V_0) \tag{13.2.10}$$

其中，U_0、V_0 为标准白光所对应的 U、V 值。

在 ULCS 体系中，色差可以用欧几里德（Euclid）距离来表示，即该色度体系可用于测量色差，两种不同颜色 C_a、C_b 间的色差 ΔC 可用下式计算：

$$\begin{aligned}\Delta C^2&=(U_a^*-U_b^*)^2+(V_a^*-V_b^*)^2+(W_a^*-W_b^*)^2\\&=(\Delta U^*)^2+(\Delta V^*)^2+(\Delta W^*)^2\end{aligned} \tag{13.2.11}$$

13.2.3 RGB 彩色模型

在 RGB 模型中，每种颜色出现在红、绿、蓝的原色光谱分量中，这个模型基于笛卡尔坐标系统，所考虑的彩色子空间是如图 13-5 所示的立方体。

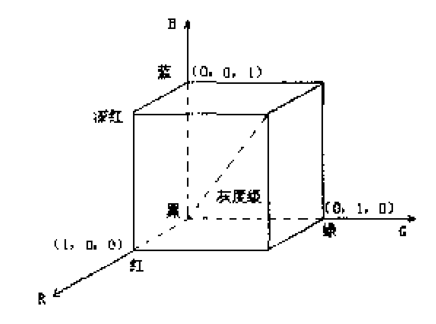


图 13-5 RGB 彩色立方体示意图

图 13-5 中，R、G、B 位于立方体的三个角上；青、品红和黄位于另外三个角上；黑色在原点处，坐标为（0，0，0）；白色位于离原点最远的角上，坐标为（1，1，1）。在该模型中，不同的颜色处在立方体上或其内部，并可用从原点分布的向量来定义。方便起见，假定所有的颜色值都被归一化，则前面图 13-5 所示的立方体就是一个单位立方体，亦即所有 R、G、B 的值都在［0，1］范围内取值。

在 RGB 彩色模型中，图像由三个图像分量组成，每一个分量图像都是其原色图像，当送入 RGB 监视器时，这三幅图像在荧光屏上混合产生一幅合成的彩色图像。在 RGB 空间，用以表示每一像素的比特数叫做像素深度。对 RGB 图像来说，其中每一幅红、绿、蓝图像都是一幅 8 比特图像，因而，每一个 RGB 彩色像素（（R，G，B）值三个一组）称为有 24bit 深度（图像平面（3）乘以每平面比特数（8）），常用"全彩色图像"来定义 24 比特的彩色图像。在 24bitRGB 图像中颜色总数是 $(2^8)^3 = 16777216$。图 13-6 显示了与图 13-5 相对应的 24bit 彩色立方体。

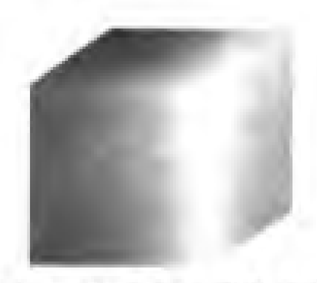


图 13-6 RGB24-bit 彩色立方体

【例 13-1】产生隐藏面和 RGB 彩色立方体的横截面。

前面图 13-5 所示的立方体是实心的，前边提到过，该立方体由 $(2^8)^3 = 16777216$ 种颜色组成。观察这些颜色的一个方便的方法是产生一个彩色平面（立方体的表面或横切面）。固定三种颜色中的一个而允许其他两个颜色变化来得到彩色平面。例如，在图 13-5 和图 13-6 中，通过立方体中心并与 GB 平面平行的横截面是（127，G，B）平面，其中 G，B =0，1，2，…，255。在这里，用的是实际像素值，而不是为方便在［0，1］范围内的数学归一化值，因为前者往往用于在计算机中产生颜色的值。图 13-7（a）显示了一幅横截面图像，该图像是把三幅独立的分量图像送入彩色监视器上观察得到的。在分量图像中 0 代表黑色，255 代表白色。

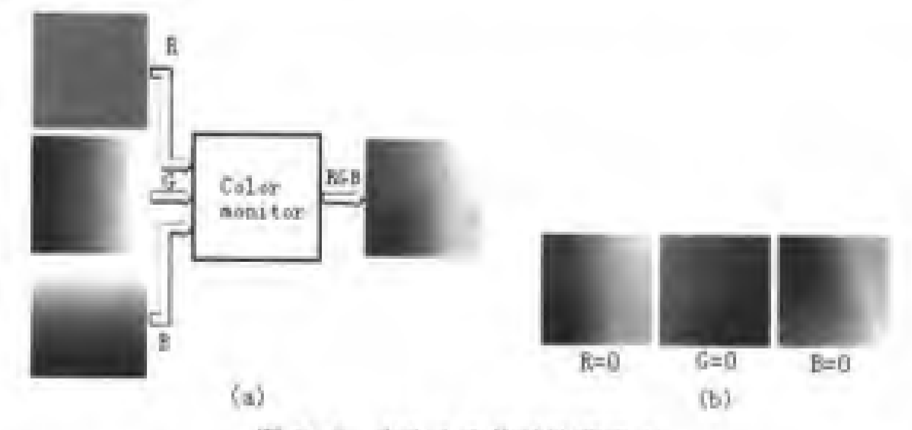


图 13-7 产生立方体的隐藏平面

注意：这些是灰度图像。图 13-7（b）显示了在图 13-5 中用同样方法产生的立方体的

三幅隐藏平面。

实际上，获得彩色图像基本上就是图 13-7 所示的相反过程。一幅彩色图像可用三个分别对红、绿、蓝敏感的滤色片得到。当用装有一种滤色镜的单色摄像机观看彩色场景时，结果是一幅单色图像，其亮度与滤色片的响应成正比。用每一个滤色片重复这一过程产生三幅单色图像，这些图像就是彩色场景的 RGB 分量图像（实际上，通常彩色图像传感器将这一过程集成在一个装置中）。图 13-7（a）所示的形式显示这三幅 RGB 分量图像就会产生一幅原彩色场景的 RGB 彩色复现图像。

虽然高端显示卡和监视器已经能够提供 24bit 的 RGB 图像，但今天所用的许多系统仍然限制在 256 种颜色。而且，有时只采用几种颜色的情况也有大量应用，在 13.6 节将讨论的伪彩色图像处理技术就是一个较好的例子。在实际应用的多种颜色系统中，有一种颜色子系统可保证颜色的真实再现，且其硬件性能与观察者无关，这种彩色子系统就是安全 RGB 彩色子集，也称作全系统安全彩色集，在网络应用中叫做无损 Web 彩色，或者叫做无损监视彩色。

假定 256 种颜色是最小颜色数，这些颜色可用任何系统真实再现。该系统可以显示所希望的彩色，用可接受的标准符号来表示这些颜色是很有用的。已知 256 种颜色中的 40 种用于各种操作系统进行不同的处理，留下的 216 种颜色则是各种系统通用的。这 216 种颜色就成为事实上的安全彩色，特别是在网络应用中，无论在何时何地，希望大多数人观察到的颜色都一样。

216 种安全彩色的每一种还是由 RGB 值形成，但每一个值只能取 0，51，102，153，204 或 255。这样，RGB 三元组数值给出 $(6^3)=216$ 种可能的值（其中每个值都可被 3 整除）。通常这些值可用 16 进制数系统来表示，如表 13-1 所示。

表 13-1 安全色中 RGB 分量的可用值

进制系统	彩色值					
16 进制	00	33	66	99	CC	FF
10 进制	0	51	102	153	204	255

16 进制数 0，1，2，…，9，A，B，C，D，E，F 对应 10 进制数 0，1，2，…，9，10，11，12，13，13，15，还可以表示为 $(0)_{16}=(0000)_2$ 和 $(F)_{16}=(1111)_2$，例如：

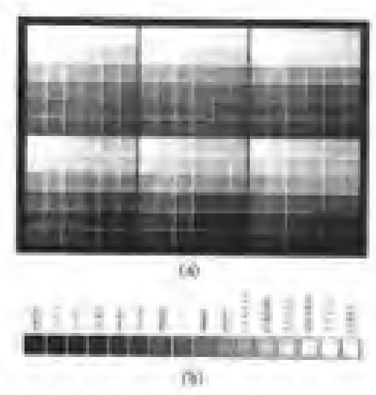
$(FF)_{16}=(255)_{10}=(11111111)_2$

两个 16 进制数组构成 8bit 的一个字节。

因为取三个数形成 RGB 彩色，每个安全色由表 13.1 中的三个两位 16 进制数形成。例如，最纯净的红色是 FF0000，黑色是 000000，白色是 FFFFFF。类似的结果也可用十进制表示得到，例如，最亮的红色在十进制表示中为 R = 255（FF），G = B = 0。

图 13-8（a）显示了由 RGB 转换而来的 216 种安全颜色，第一行左上角方块数值是 FFFFFF（白），它右边的方块数值为 FFFFCC，第三块为 FFFF99，等等。同一阵列的第二行的值是 FFCCFF，FFCCCC，FFCC99 等等。阵列中最后一个方块为 FF0000，第二阵列的右边正好由 CCCCCC 开始，并以相同的方式前进，剩下的四个阵列也是一样，最后一个阵列的最后一个方块（底部右方）为 000000（黑）。需注意的是，并不是所有 8bit 灰色都包含在 216 种安全色中。图 13-8（b）显示了在 256 色 RGB 系统中，对所有可能灰色的 16 进制编码。这些值的一部分在安全色集的外边，但在多数显示系统中可正常表示（根据它

们的相对强度)。来自安全彩色组的灰度$(KKKKKK)_{16}$，$K=0$，3，6，9，C，E，如图13-8(b)的最下边所示。



(a)

(b)

图13-8 216种安全颜色

图13-9显示了RGB安全彩色立方体，不像图13-6所示的全彩色立方体是实心的，图13-9的立方体仅在表面有效。正如图13-8(a)所示，每一平面都有36种颜色，所以安全彩色立方体的所有表面由216种不同的颜色覆盖。

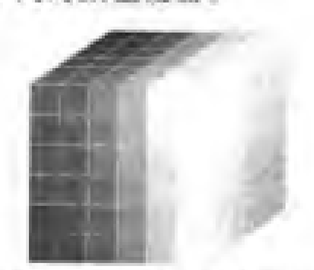


图13-9 RGB安全彩色立方体

13.2.4 CMY、CMYK彩色模型

青、品红、黄(CMY)分别是红、绿、蓝(RGB)的补色，这三色通常被称为减色基色，而红、绿、蓝则称为加色基色。CMY彩色模型被广泛应用于彩色图像印刷设备，如彩色打印机、彩色复印机等，这些设备都要求输入CMY数据或在内部作RGB到CMY的转换，RGB到CMY的转换按以下公式进行：

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{13.2.12}$$

这里，假设所有的颜色值都归一化为[0，1]范围。式(13.2.12)显示了从涂有青色颜料的表面反射的光不包含红色(即公式中C=1-R)。与此相似，纯品红色不反射绿色，纯黄色不反射蓝色。式(13.2.12)还显示RGB值可以很容易地通过1减去CMY值从CMY集中得到，在图像处理中，这一模型主要用于产生硬拷贝输出，因此，从CMY到RGB反向操作通常没有实际意义。

根据图13-3所示，等量的颜料原色(青、品红和黄色)可以产生黑色。而实际上，由于打印机中所用的墨水不可能是理想的，因而产生的黑色是不纯的。因此，为了产生真正的黑色(黑色在打印中往往起主要作用)加了第四种颜色——黑色，提出了CMYK彩色模型：

$K = \min(C, M, Y)$

$C = C - K$

$M = M - K$

$Y = Y - K$

其中 K 表示黑色，因此，当出版商说到“四色打印”时，是指彩色模型的三种减色基色再加上黑色。

13.2.5 HSI 模型

当观察一个彩色物体时，往往用色调、色饱和度和亮度描述它。Munseu 根据肉眼的色彩视觉三要素：色调（Hue）、饱和度（Saturation）和亮度（Intensity）提出了 HSI 彩色方案，也即 HSI 彩色模型。

HSI 模型中的 I 表示强度或亮度，即肉眼感受到光的明暗程度，光波能量增大，亮度就增大；反之能量减弱，亮度则减小。实际上，强度（或称亮度）是 R、G、B 三个灰度值的平均值，它直接决定像素的整体亮度。

HSI 模型中的 H 表示色调，即光的不同颜色，如红、橙、黄、绿、青、蓝、紫分别表示不同的色调。不同的波长呈现不同的颜色，就是指不同的色调。发光物体的色调取决于它产生的辐射光谱的分布。

HSI 模型中的 S 表示饱和度，即彩色的深浅程度。饱和度的深浅与色光中白光成分的多少有关。一种纯彩色光中加入的白光成分越少，该彩色的饱和度越高；反之，白光成分越多，饱和度就越低。因而，饱和度反映了某种色光被白光冲淡的程度。

HSI 模型对于开发基于彩色描述的图像处理方法是一个理想的工具，这种彩色描述更自然、直观。而 RGB 对图像彩色的产生（用彩色摄像机捕获图像或者把图像显示在监视器屏幕上）是理想的，但在彩色描述上有较多的限制。

例 13-1 通过用三幅单色灰度图像来观察 RGB 彩色图像（描述红、绿和蓝），同样，也可以将强度从 RGB 图像中提取出来。

用前面图 13-5 的彩色立方体并置黑顶点为（0，0，0），白顶点为（1，1，1）。正如图 13-7 指出的，强度（灰度级）沿着连接两顶点的连线分布，如图 13-10 所示，连接黑和白顶点的线垂直于水平面，这样，如果要确定图 13-10 中任何点的强度分量，可以做一个通过该点且垂直于强度轴的平面，该平面与强度轴的交点就给出了在［0，1］范围内的强度值。彩色的饱和度（纯度）随着该彩色与强度轴的距离的增加而增加。通过观察容易知道，强度轴上点的饱和度是零，沿着这条轴线的所有点都是灰度。

为了了解如何从给定的 RGB 点确定色调，考虑图 13-10（b），图中显示了由三个点（黑、白和青）定义的平面，黑点和白点包含在平面内，所以强度轴也在平面内。而且，位于由强度轴和立方体边界确定的平面内的所有点都有相同的色调（青色）。因此可以得到这样的结论，即所有颜色都是由位于那些颜色定义的三角形内的三种颜色产生的。如果这些点的两点是黑和白，第三点是彩色点，在三角形上的所有点都有相同的色调，因为黑和白分量不能改变色调（当然在这个三角形中点的强度和饱和度是不同的）。沿着垂直强度轴旋转色调平面，可得到不同的色调。因此可得到这样的结论，HSI 空间要求的色调、饱和度和强度可以从 RGB 彩色立方体得到。也就是说，可以把任何 RGB 点转换为相应的 HSI 彩色模

型的点。

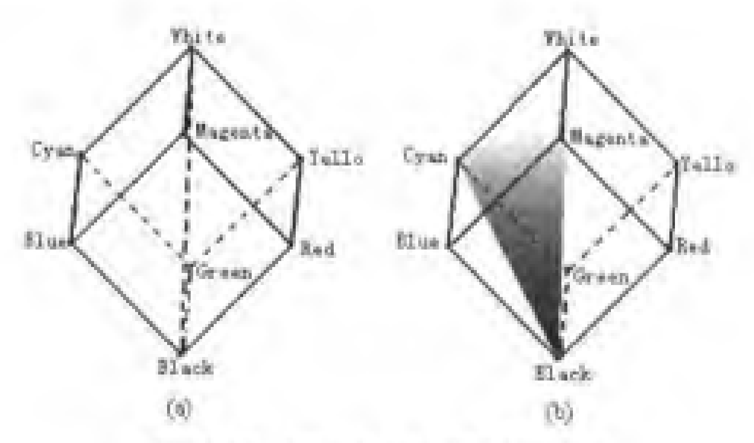


图 13-10 RGB 和 HSI 模型间的关系

关于图 13-10 所示的立方体结构和相应 HSI 空间的关键一点是，垂直强度轴和位于垂直于该轴的平面的彩色点轨迹表示 HSI 空间。当平面沿强度轴向上或向下移动时，由平面与立方体表面构成的横截面决定的边界不是呈三角形就是呈六边形。图 13-11（a）表示了横截面为六边形的情况，在立方体沿灰度轴向下看，就会清楚得多。

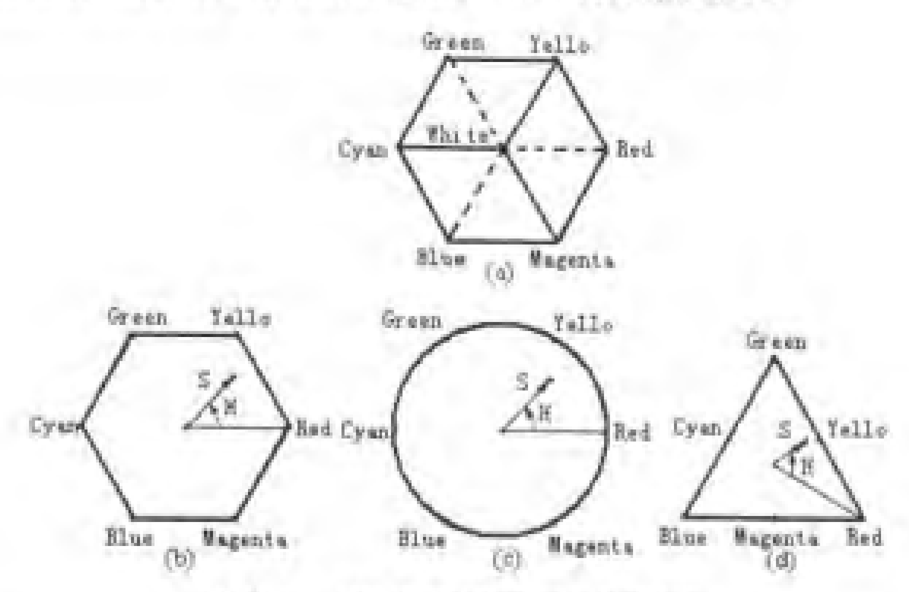


图 13-11 HSI 模型中的色调和色饱和度

在这个平面中可以看到原色是按120°分隔的，二次色与原色相隔60°，这意味着二次色之间也相隔120°。图 13-11（b）显示了与 13-11（a）相同的六边形和其上的任意一个彩色点（用一点显示）。该点的色调由来自某参考点的一个角度来决定，通常（不总是）将与红轴（原点到 Red 点连线）成0°角的向量上的点指定为 0 色调，从这里开始色调逆时针增长。饱和度（距垂直轴的距离）是从原点到该点的向量长度。

注意：原点是由彩色横截面与垂直强度轴定义的。HSI 彩色空间的重要分量是垂直强度轴到一彩色点的向量长度和这一向量与红轴的角度。因此，如图 13-11（c）和（d）所示，HSI 平面以前面讨论的六边形、三角形，甚至一个圆形的形式出现并不奇怪。实际上，选择什么形状并不重要，因为这些形状中的任何一个都可以通过几何变换变为其他两种。

图 13-12 显示了基于彩色三角形和圆形的 HSI 模型。图中的黑点（用 S 表示）是一个

任意彩色点。与红轴的夹角给出色调，向量的长度是饱和度，在该平面中，所有彩色的强度由平面在垂直强度轴的位置给出。

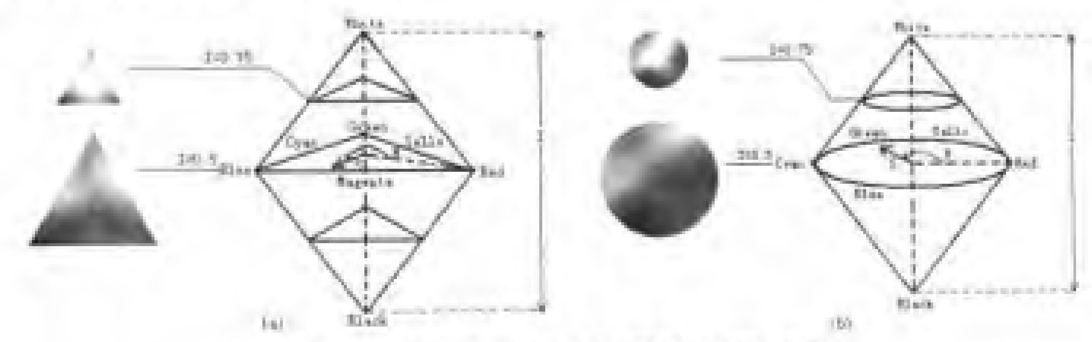


图 13-12 基于彩色三角形和圆形的 HSI 模型

13.2.6 RGB 到 HSI 彩色模型的转换

一幅给定的 RGB 彩色格式的图像，根据每一个 RGB 像素可以得到 H 分量：

$$H=\begin{cases}\theta\ , & B\leqslant G\\ 360-\theta\ , & B\geqslant G\end{cases} \tag{13.2.13}$$

其中：

$$\theta=\arccos\left\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-G)(G-B)]^{\frac{1}{2}}}\right\}$$

色饱和度分量：

$$S=1-\frac{3}{(R+G+B)}[\min(R,G,B)] \tag{13.2.14}$$

强度分量：

$$I=\frac{1}{3}(R+G+B) \tag{13.2.15}$$

就像图 13-11 指出的那样，假定 RGB 值归一化为［0，1］范围内，角度 θ 根据 HSI 空间的红轴来度量，色调可以用式（13.2.13）得到的值除以 360°归一化为［0，1］范围内，那么只要给出的 RGB 值在［0，1］区间内，则得到的其他两个 HSI 分量一定在［0，1］范围内。

13.2.7 HSI 到 RGB 彩色模型的转换

在［0，1］内给出 HSI 值，要在相同的值域找到 RGB 值，可利用求 H 值的公式。在原始色分割中有三个相隔120°的扇形（见图 13-11），将 H 乘以 360°，色调值又落回到[0°，360°]的范围。

以下分析 H 值落入下列各扇区时 HSI 到 RGB 的转换：

RG 扇形（$0°\leqslant H<120°$）：当 H 位于这个扇形区时，可通过下式得到 RGB 分量：

$$B=I(1-S) \tag{13.2.16}$$

$$R=I[1+\frac{S\cos H}{\cos(60°-H)}] \tag{13.2.17}$$

$$G = 1-(R+B) \tag{13.2.18}$$

GB 扇区（$120^\circ \leqslant H < 240^\circ$）：如果给定的 H 的值在这一扇区，首先，将 H 的值减去 120°：

$$H = H - 120^\circ \tag{13.2.19}$$

然后求得 RGB 分量：

$$R = I(1-S) \tag{13.2.20}$$

$$G = I[1+\frac{S\cos H}{\cos(60^\circ - H)}] \tag{13.2.21}$$

$$B = 1-(R+G) \tag{13.2.22}$$

BR 扇区（$240^\circ \leqslant H < 360^\circ$）：如果 H 落在这一扇区，则先将 H 的值减去 240°：

$$H = H - 240^\circ \tag{13.2.23}$$

求得 RGB 分量：

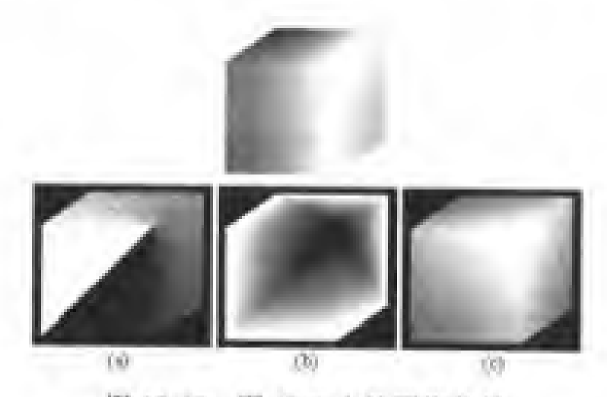
$$G = I(1-S) \tag{13.2.24}$$

$$B = I[1+\frac{S\cos H}{\cos(60^\circ - H)}] \tag{13.2.25}$$

$$R = 1-(G+B) \tag{13.2.26}$$

【例 13-2】 对应于 RGB 立方体图像的 HSI 值。

图 13-13 显示了 RGB 彩色立方体图像的色调、饱和度和强度。图 13-13（a）是色调图像，其最大特点是，在立方体前平面沿 45° 线的值是不连续的；图 13-13（b）中，饱和度图像显示出从暗值渐进地向 RGB 立方体的白顶点过渡，表明彩色饱和度越来越小，一直逼近白色；图 13-13（c）的强度图像中的每一个像素值等于 RGB 彩色立方体中相应像素 RGB 的平均值。



(a)　(b)　(c)

图 13-13　图 13-6 中的图像分量

13.3 彩色变换

在本节中将要讨论的彩色变换，不是指模型间的分量的转换（像前面提到的 RGB 到 HSI，HSI 到 RGB 的转换），而是指在单一彩色模型的范围中，对彩色图像分量进行变换处理。

13.3.1 公式

以下式作为彩色变换的模型：

$$g(x,y) = T[f(x,y)] \tag{13.3.1}$$

其中，$f(x,y)$指彩色输入图像，$g(x,y)$是经变换或处理过的彩色输出图像，T是对f的操作。这里，像素值是从彩色空间选择的用来描述图像的3元组或4元组。可用下式表示彩色变换：

$$s_i = T_i(r_1, r_2, \ldots, r_n) \qquad i=1,\ 2,\ \cdots,\ n \tag{13.3.2}$$

其中，r_i和s_i是$f(x,y)$和$g(x,y)$在某点的彩色分量的变量，$\{T_1, T_2, \ldots, T_n\}$是一个对$r_i$操作产生$s_i$的变换或彩色映射函数集，n个变换$T_i$合并执行式（13.3.1）中的单一变换函数$T$；n的值由所选择的用于描述$f$和$g$像素的彩色空间决定，例如，选择RGB彩色空间，则n=3，r_1，r_2和r_3分别表示输入图像的红、绿、蓝分量；选择CMYK或HSI彩色空间，则n=4或n=3。

图13-14的第一行显示了一碗草莓和一个咖啡杯的高分辨率彩色图像。

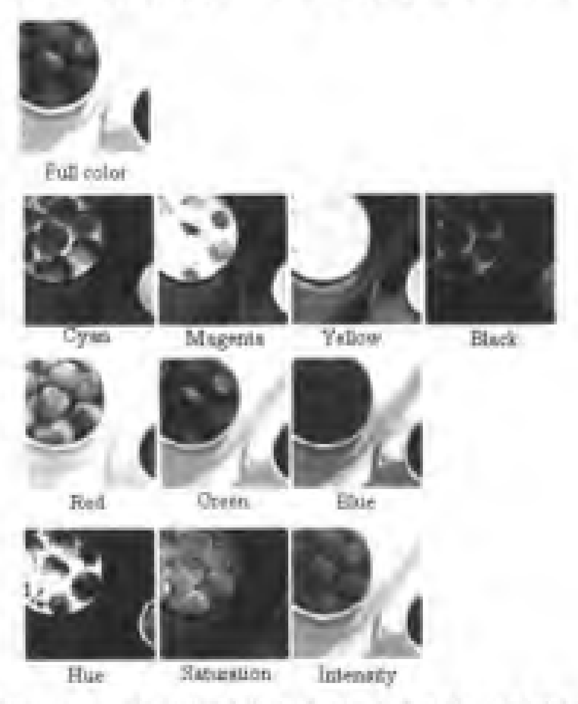


图13-14 全彩色图像和它的不同彩色空间分量

图13-14中的第二行是原始的CMYK扫描分量图像，在这些图像的每一个CMYK彩色分量中，白用1表示，黑用0表示。可以看到对应于品红和黄色这两个分量的图像最亮，也就是说，草莓是由大量的品红和黄色组成的；黑色较少并通常限于咖啡和草莓碗中的阴影。当CMYK图像被转换为RGB时（图13-14中第三行），可以看到草莓包含大量的红色和很少的绿与蓝色。图13-14的最后一行显示了用式（13.2.13）到式（13.2.15）计算出的图13-14中原彩色图像的HSI分量像。强度分量是全彩色原像的单色复现，另外，草莓在彩色方面相对较纯净，它们具有最高的饱和度或者说它们的色调被白光稀释得最少。然而，对色度分量进行说明时还有些困难，原因如下：

（1）在HSI模型中，0°和360°相遇处有一个不连续点。

（2）色调对于0饱和度没意义（对白、黑和纯灰）。

模型的不连续点多出现在草莓周围，它们用接近白（1）和黑（0）的灰度值描述，也即用高对比灰度级的混合去描述单一颜色——红色，这是人们不希望的。

图13-14中的任何彩色空间分量都可与式（13.3.2）相配合。理论上，任何变换都可在

任何模型中执行，但实际应用中发现，某些操作对特定的模型比较适用。对给定的变换，有时需要先将彩色模型转换到该变换更加适用的彩色空间，然后再在该空间执行转换。例如，假设要改进图 13-14 中原彩色图像的亮度，使用：

$$g(x,y)=kf(x,y) \tag{13.3.3}$$

这里 $0<k<1$。在 HSI 彩色空间，可以做简单变换：

$$s_3=kr_3 \tag{13.3.4}$$

其中，$s_1=r_1$，$s_2=r_2$，只是 HSI 亮度分量 r_3 被改变。

在 RGB 彩色空间，3 个分量都需要进行变换：

$$s_i=kr_i \quad i=1,2,3 \tag{13.3.5}$$

在 CMY 空间则需要一个相似的线性变换集：

$$s_i=kr_i+(1-k) \quad i=1,2,3 \tag{13.3.6}$$

虽然 HSI 变换包含最少的操作，然而，把 RGB 或 CMYK 图像转换为 HSI 空间却需要比较复杂的计算，大大抵消了简单变换的优点，即转换计算量比亮度变换本身计算量大得多。然而，不论选择哪一个彩色空间，只要变换方程相同，其输出是相同的。图 13-15（b）显示了采用式（13.3.4）到式（13.3.6）对图 13-13 中原彩色图像用 $k=0.7$ 进行变换的结果。映射函数如图 13-15（c）～（e）所示。

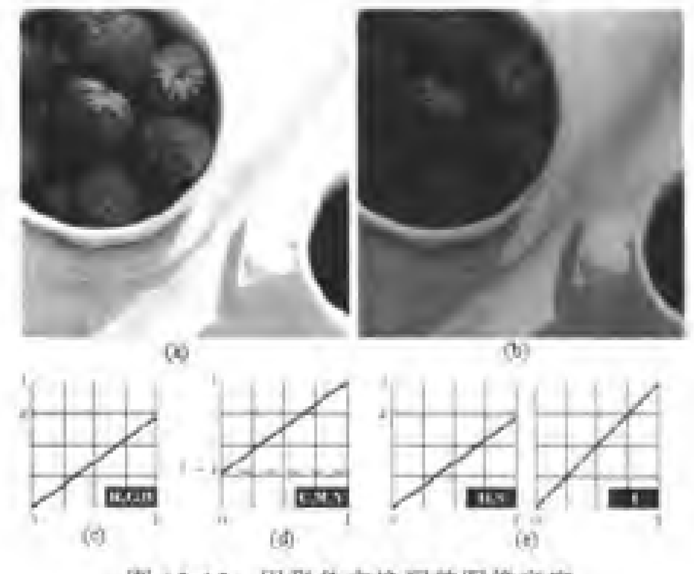


图 13-15　用彩色变换调整图像亮度

（a）原图像。

（b）减少亮度 30% 的结果（k = 0.7）。

（c）～（e）所要求的 RGB、CMY 和 HSI 变换函数。

注意：式（13.3.4）到式（13.3.6）所定义的每一变换仅仅依赖于其彩色空间的一个分量。例如，红的输出分量 S，在式（13.3.5）中独立于绿和蓝输入，它只依赖于红输入。所以说，这类变换是最简单的和最常用的彩色处理工具，并可以以每个彩色分量为基础进行。

13.3.2　补色

在如图 13-16 所示的彩色环上，与一种色调相对立的另一种色调称为其补色。补色的概念有些类似于灰度负值，它对于增强嵌在彩色图像暗区的细节，特别是增强在大小上占

支配地位的细节方面很有用。

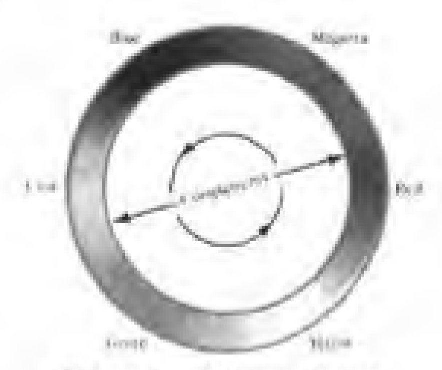


图 13-16 彩色环上的补色

【例 13-3】计算彩色图像的补色。

图 13-17（a）和（c）显示了来自图 13-13 的彩色图像和它的补色图像。用于计算补色的 RGB 变换如图 13-17（b）所示，计算补色有些类似于照片的彩色负片。原图像的红色在补色中用青色代替，当原始图像是黑色时，补色是白色等等。补色图像中的每种色调都可从原图像用图 13-16 的彩色图来预测。涉及补色计算的每一个 RGB 分量变换仅是对应的输入彩色分量的一个函数，不像图 13-15 的亮度变换，在该例子中使用的 RGB 补色变换函数没有直接的 HSI 空间等效当量。图 13-17（d）提供了一种使用图 13-17（b）给出的色调、饱和度和亮度变换的补色的一种近似。

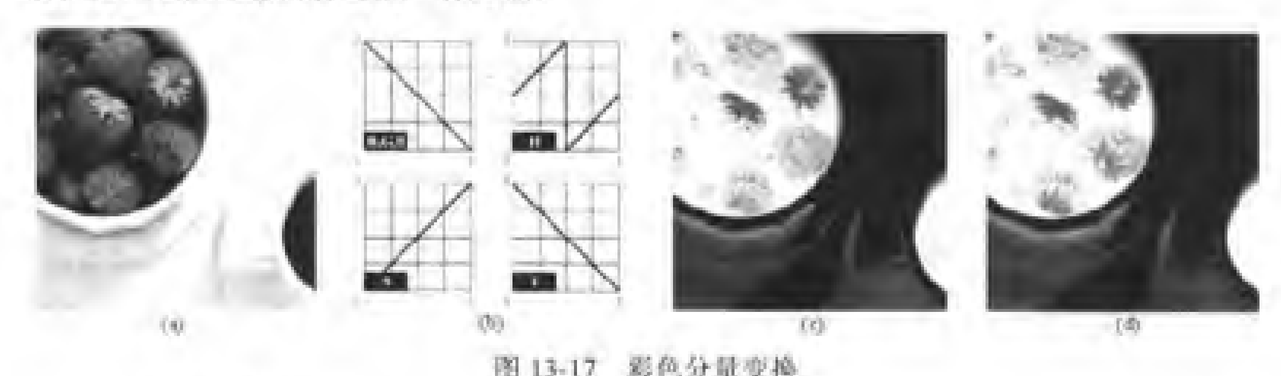


图 13-17 彩色分量变换

（a）原图像。

（b）补色变换函数。

（c）基于 RGB 映射函数的（a）的补色。

（d）用 HSI 变换的 RGB 补色的近似。

注意：输入图像的饱和度分量是不可改变的。

13.3.3 彩色分层

彩色分层的目的是突出图像中特殊的彩色区域，这一步骤对分离出目标物非常有用。其基本思路如下：

（1）显示感兴趣的颜色，以便从背景中把它们分离出来。

（2）像模板那样使用由彩色定义的区域，以便进一步处理。

当然，最直接的方法是沿用灰度分层技术，然而，由于一个彩色像素是一个 n 维参量，所以彩色变换函数比相对应的灰度变换函数要复杂得多。事实上，彩色变换所要求的变换

比到目前为止考虑的彩色分量变换也复杂得多，因为所有的彩色分层方法都要求每个像素变换后的彩色分量是所有 n 个原始像素彩色分量的函数。对一幅彩色图像分层最简单的方法之一是把某些感兴趣区域以外的区域的彩色映射为不突出的自然色。如果感兴趣的颜色由宽为 W，中心在原彩色（平均）点并具有分量（$a_1,a_2,\ldots,a_n$）的立方体（或超立方体，n>3）所包围，则必要的变换集是：

$$s_i=\begin{cases}0.5\ , & \left[\left|r_j-a_j\right|>\dfrac{W}{2}\right]_{\text{any } 1\le j\le n} \\ r_i\ , & \text{其他}\end{cases},\quad i=1,2,\cdots,n \tag{13.3.7}$$

在这些变换中，强迫其他颜色为参考彩色空间的中点（任意选取的中间点），以突出原型周围的颜色。例如，对于 RGB 彩色空间，灰度或彩色的中点（0.5 ，0.5，0.5）可以作为一个合适的自然点。若用一个圆球确定感兴趣的颜色，式（13.3.7）变为：

$$s_i=\begin{cases}0.5\ , & \sum_{j=1}^{n}(r_j-a_j)^2>R_0^2 \\ r_i\ , & \text{其他}\end{cases},\quad i=1,2,\cdots,n \tag{13.3.8}$$

其中，R_0 是封闭球形的半径（或超级球形，n>3），（$a_1,a_2,\ldots,a_n$）是其中心的分量（原彩色）。式（13.3.7）和式（13.3.8）另外还可以进行的变化，包括：实现多彩色模型和在感兴趣之外的区域减小彩色亮度，而不是赋以中间值。

【例 13-4】 对彩色分层的一种说明。

将式（13.3.7）和式（13.3.8）用于图 13-13 中的原彩色图像，可以分离出草莓食品。图 13-18（a）和（b）即通过两种变换得到的结果。



(a)



(b)

图 13-18 彩色分层

（a）在宽度为 W = 0.2549，中心点在（0.6863，0.1608，0.1922）的 RGB 立方体中检测红色的彩色分层变换。

（b）在半径为 0.1765，中心在相同点的 RGB 球形中检测红色，处在立方体和球形外边的像素由彩色（0.5，0.5，0.5）代替。

13.3.4 彩色平衡

彩色图像经过数字化后，色通道中不同的敏感度、增光因子、偏移量等因素导致图像的三个分量出现不同的线性变换，使得图像的三基色不平衡，造成图像中物体的颜色偏离原有的真实色彩，最突出的现象是使灰色的物体着了伪色。检查彩色是否平衡的最简单的方法是看图像中原为灰色的物体是否仍然是灰色，高饱和度的颜色是否有正常的色度。如果图像有明显的黑色或白色背景，在 RGB 分量的直方图中会产生显著的峰，但如果各个

直方图中的峰处在三基色不同的灰度级上，则表明彩色出现不平衡。这种不平衡可通过对R，G，B三个分量分别使用线性灰度变换进行纠正。一般只需要变换分量图像中的两个与第三个匹配即可，最简单的灰度变换函数的设计方法如下：

（1）选择图像中相对均匀的浅灰和深灰两个区域。

（2）计算这两个区域的三个分量图像的平均灰度值。

（3）调节其中两个分量图像，用线性对比度使其与第三幅图像匹配。

如果所有三个分量图像在这两个区域中具有相同的灰度级，则完成了彩色平衡调节。

13.3.5 直方图处理

因为彩色图像是由多个分量组成的，所以必须考虑适应多于一个分量的直方图的灰度级技术。对彩色图像的各个分量独立的进行直方图均衡是行不通的，往往会产生不正确的彩色。一个比较合乎逻辑的方法是均匀地扩展彩色强度，保留颜色本身（也即色调）不变，HSI空间是适合该种情况的理想方法。

【例13-5】在HSI彩色空间的直方图均衡。

图13-19（a）显示了包括调味瓶和摇杯的调味瓶架台的彩色图像，其强度分量的范围值为［0，1］（归一化后的）。由处理前的强度分量直方图（图13-19（b））中看到的，图像包含大量的暗彩色，使中央强度减少到0.36。不改变色调和饱和度，均衡强度分量的结果示于图13-19（c）。容易发现，所有图像有效的加亮了，并且一些调味瓶和放调味瓶的木桌纹理都能看到了，图13-19（b）显示了新图像的直方图以及用于均衡强度分量的变换。

虽然强度均衡处理并未改变图像的色调和饱和度值，但它的确影响了整体图像的彩色感观。图13-19（d）显示了采用增加图像的饱和度分量，然后利用图13-19（b）所示的变换进行直方图均衡这一特殊校正的结果，这类调整在HSI空间处理强度分量时非常适用，因为强度改变通常影响到图像的彩色状态。

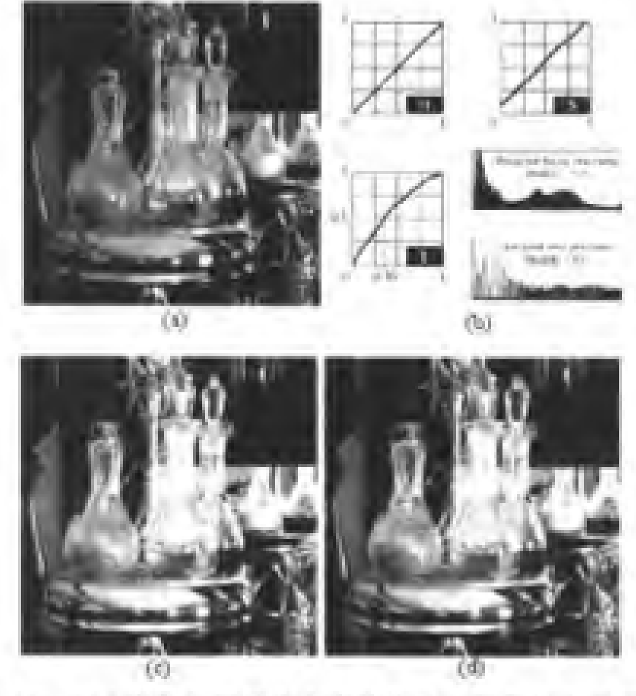


图13-19 HSI彩色空间的直方图均衡（后面是饱和度调整）

13.4　平滑和锐化

13.4.1　彩色图像的平滑

在对灰度图像进行平滑时，可以将其看作是空间滤波处理，即将系数都是 1 的模板滑过图像，每一个像素由模板定义的邻域中像素的平均值代替。这一概念也可以扩展到彩色图像处理中，其主要差别在于不再是处理灰度标量值，而变为处理彩色像素的分量向量。

令 S_{xy} 表示在 RGB 彩色图像中定义的一个中心在 (x,y) 的邻域的坐标集，在该邻域中 RGB 分量的平均值为：

$$\bar{c}(x,y)=\frac{1}{K}\sum_{(x,y)\in S_{xy}}c(x,y) \tag{13.4.1}$$

向量的附加特性为：

$$\bar{c}(x,y)=\begin{bmatrix}\frac{1}{K}\sum_{(x,y)\in S_{xy}}R(x,y)\\ \frac{1}{K}\sum_{(x,y)\in S_{xy}}G(x,y)\\ \frac{1}{K}\sum_{(x,y)\in S_{xy}}B(x,y)\end{bmatrix} \tag{13.4.2}$$

显然，对彩色图像进行平滑，就是对 RGB 图像的每一个平面用传统的灰度邻域处理方法分别进行平滑。于是，得到这样一个结论：用邻域平均值平滑可以在每个彩色平面的基础上进行，其结果与用 RGB 彩色向量执行平均是相同的。

【例 13-6】用邻域平均对彩色图像进行平滑。

图 13-20（a）为彩色 Lena 图像，图 13-20（b）~（d）分别是该图像的红、绿、蓝平面。



图 13-20　邻域平均法平滑彩色图像

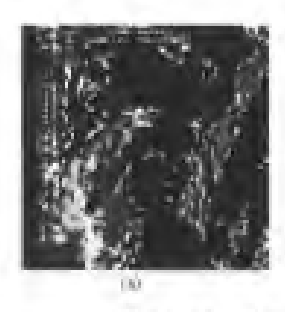
（a）RGB 图像。

（b）红分量图像。

（c）绿分量图像。

（d）蓝分量图像。

图 13-21（a）～（c）显示了图像的 HSI 分量。正如前面提到的，可以用 5×5 的灰度平均模板对图 13-20 的彩色图像进行平滑，单独平滑 RGB 彩色平面的每一个，然后混合处理过的平面，就可以形成一幅平滑的全彩色结果，按照这种方法计算过的图像，如图 13-22（a）所示。


(a)

(b)

(c)

图 13-21 图 13-20 中 RGB 彩色图像的 HSI 分量

（a）色调。

（b）饱和度。

（c）强度。

HSI 彩色模型的重要优点是消除了强度和彩色信息的关系（与灰度关系密切），使得许多灰度处理技术适合彩色处理，而且对强度分量平滑更有意义。为了说明这种方法的优点和重要性，下边仅对强度分量进行平滑（色调和饱和度分量保持不变）并把处理结果变换成 RGB 图像并显示于图 13-22（b）。图 13-22（c）给出了图 13-22（a）与图 13-22（b）的差别，这是由于两个不同颜色的像素平均是指两种彩色的混合，而不是原色混合，仅对强度平滑而原色调和饱和度保持不变则保留了像素的原彩色。


(a)

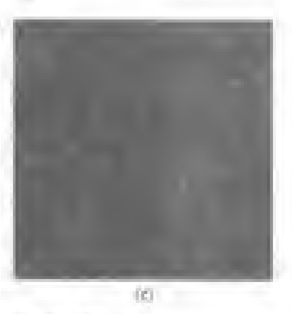
(b)

(c)

图 13-22 用 5×5 平均模板平滑图像

（a）处理 RGB 每一分量图像的结果。

（b）处理 HSI 图像强度分量，转换为 RGB 图像的结果。

（c）两种结果间的差别。

注意：差别将随着平滑模板的增加而增加。

13.4.2 彩色图像锐化

在这里，将考虑用拉普拉斯（Laplacian）方法对彩色图像进行锐化。由向量分析可知，

向量的拉普拉斯变换也为向量，其分量等于输入向量的各独立标量分量的拉普拉斯微分。

在RGB彩色系统中，向量$\vec{c}$的拉普拉斯变换为：

$$\nabla^2[\vec{c}(x,y)]=\begin{bmatrix}\nabla^2 R(x,y)\\ \nabla^2 G(x,y)\\ \nabla^2 B(x,y)\end{bmatrix} \tag{13.4.3}$$

由此，可以通过分别计算图像的每一分量的拉普拉斯变换来计算整个彩色图像的拉普拉斯变换。

【例13-7】用拉普拉斯进行锐化。

图13-23（a）是对RGB每个分量进行拉普拉斯变换，并混合它们产生的锐化结果。图13-23（b）显示了基于图13-21的HSI分量的锐化结果，该结果是将强度分量的拉普拉斯变换和色调、饱和度相混合后得到的结果。RGB和基于HSI的结果间的差别，如图13-23（c）所示，产生差别的原因与例13-6中的相同。



图13-23　用拉普拉斯进行图像锐化

（a）处理每一个RGB通道的结果。

（b）处理强度分量并转换为RGB的结果。

（c）两种结果的差别。

13.5　彩色分割

13.5.1　HSI彩色空间分割

若要对一幅彩色图像进行分割，而且要在单独的平面上执行处理，会很自然地想到HSI空间，因为在色调图像中描述彩色是非常方便的。人们往往可以将饱和度作为一个模板图像，然后在色调图像中分离出感兴趣的区域。下面的例子说明了如何在HSI空间中对图像进行分割。

【例13-8】在HSI空间对彩色图像进行分割。

人们的兴趣是分割图像13-24（a）中左下角的微红色区域。虽然它是用伪彩色产生的，但是用处理全彩色图像的方法处理通常不会有损失。图13-24（b）～（d）是它的HSI分量图像。比较图13-24（a）和图13-24（b）可知，人们感兴趣的区域有较高的色调值，表明该彩色在红色的蓝及品红一侧（图13-11）。图13-24（e）显示了根据阈值产生的二值图像，该图像在饱和度图像中的阈值等于最大饱和度的10%，任何大于门限值的像素赋1值（白），其他赋0值（黑）。图13-24（f）用色调图像产生模板，图13-24（g）是结果图像

的直方图（灰度尺度在［0，1］范围内）。在直方图中，高值（这是感兴趣的值）被分组到灰度标尺的最高端，接近1.0。以0.9为阈值的阈值化图像产生的二值图像，如图13-24（h）所示。在该图像中，根据白点的空间位置识别原像点，原像点具有感兴趣的微红色调。这个结果与完善的分割相差甚远，因为在原图像中还有一些有微红色调的点没有被该分割方法识别出来。然而实验表明图13-24（h）所示的白色区域在识别原图像的微红色分量时，是这种方法能做到的最好结果。在下面一节讨论的分割方法将能够产生更加好的结果。

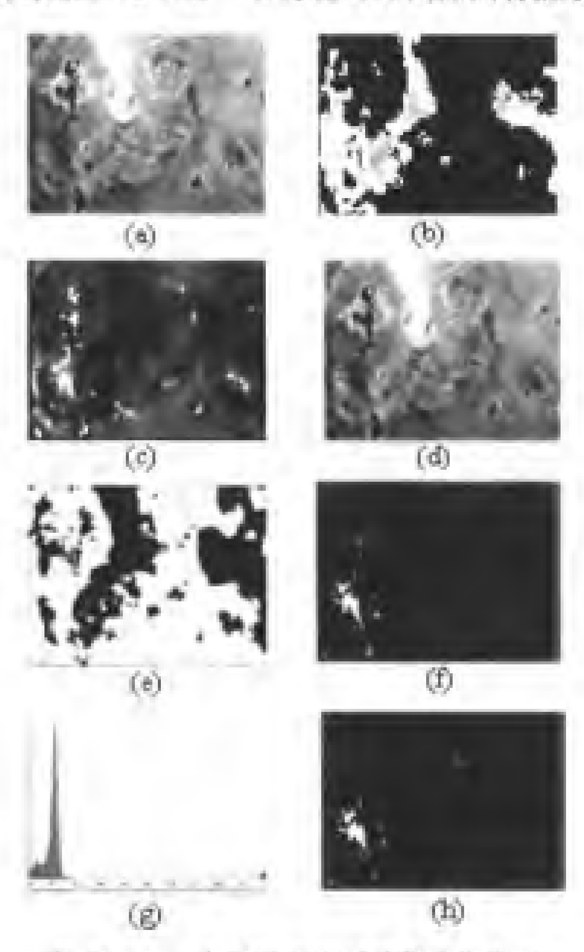


图13-24　在HSI空间对图像进行分割

（a）原图像。
（b）色调。
（c）饱和度。
（d）强度。
（e）二值饱和度模板（黑＝0）。
（f）（b）和（e）相乘。
（g）（f）的直方图。
（h）（a）中红分量的分割。

13.5.2　RGB彩色空间分割

虽然在HSI空间对彩色图像的处理更直观，但当进行分割操作时，在RGB彩色空间进行处理的会得到更好的效果。

假设要将一幅 RGB 图像中具有某特殊彩色的区域分割出来。给定一个感兴趣颜色的有代表性的颜色点样品集，可得到一个颜色"平均"估计，这种颜色就是人们希望分割出来的颜色，用 RGB 向量 $\vec{a}$ 来表示这个平均彩色。

分割的目标是对给定图像中每一个 RGB 像素进行分类，因为一个 RGB 像素要么属于这种颜色，要么不属于这种颜色。因此，为了进行比较，需要一个相似性度量，最简单的度量便是用欧氏距离进行度量。

令 $\vec{z}$ 代表 RGB 空间中的任意一点，如果 $\vec{z}$ 与 $\vec{a}$ 之间的距离小于特定的阈值 D_0，则认为二者是相似的。$\vec{z}$ 和 $\vec{a}$ 之间的欧氏距离由下式给出：

$$D(\vec{z},\vec{a})=\left\|\vec{z}-\vec{a}\right\|=\left[(\vec{z}-\vec{a})^T(\vec{z}-\vec{a})\right]^{\frac{1}{2}}=\left[(z_R-a_R)^2+(z_G-a_G)^2+(z_B-a_B)^2\right]^{\frac{1}{2}} \quad (13.5.1)$$

这里，注脚 R，G，B 表示向量 $\vec{z}$ 与 $\vec{a}$ 的 RGB 分量。$D(z,a)\leqslant D_0$ 的点的轨道是半径为 D_0 的实心球，如图 13-25（a）所示。包含在球内部和表面上的点符合特定的彩色准则，球以外的点则不符合准则。在图像中对这两类点集编码，比如用黑或白，则产生一幅二值分割图像。

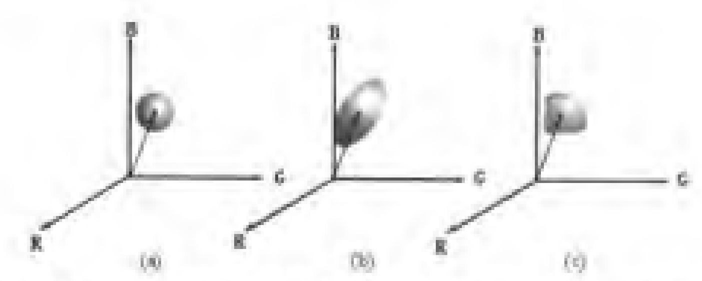


图 13-25　对于 RGB 向量分割，封闭数据范围的三种方法

式（13.5.1）一个有用的推广是如下形式的距离测度：

$$D(\vec{z},\vec{a})=[(\vec{z}-\vec{a})^T C^{-1}(\vec{z}-\vec{a})]^{\frac{1}{2}} \quad (13.5.2)$$

其中，C 是希望分割的颜色典型的样本协方差矩阵。$D(z,a)\leqslant D_0$ 的点的轨道描述了一个实心的椭球体（图 13-25（b）），它的最大特点是，主轴在最大数据范围的方向旋转。当 C = I 时，3×3 单位矩阵（式（13.5.2））简化为式（13.5.1）。分割如前一节描述的一样。

距离往往是正的且单调，所以可以用距离的平方运算来代替距离以避免开方运算。然而，即使不计算平方根，式（13.5.1）或式（13.5.2）的计算量也很大，比较折中的方案是使用边界盒，如图 13-25（c）所示。在该方法中，盒的中心在 $\vec{a}$ 上，沿每一彩色轴的尺度与沿每一个轴取样的标准差成比例，标准差的计算只使用一次样本彩色数据。对于任意一个彩色点，可以根据它是否在盒子表面或内部来进行分割，与用距离方法一样。但判定一个彩色点是在盒子内部还是外部的计算比求球或椭球界限时要简单得多。实际上，上述讨论是 13.3.3 节讨论的彩色图像分割方法的扩展。

图 13-26（a）所示的方形区域包含微红色样品，这里希望分割出该颜色，与在例 13-8 中的分割目标相同，但是在这里，利用 RGB 彩色向量对该问题进行研究。后面的方法利用包含在图 13-26（a）中长方形中的彩色点计算平均向量 $\vec{a}$，然后计算这些样点的红、绿、蓝的标准差。盒子的中点在 $\vec{a}$，它的尺度沿每一个 RGB 轴以沿相应轴的数据标准差的 1.25

倍选择。例如，令σ_R代表样点红分量的标准差，然后，盒子的尺度沿红轴从$(a_R-1.25\sigma_R)$扩展到$(a_R+1.25\sigma_R)$，这里σ_R代表平均向量$\vec{a}$的红分量。在整个彩色图像中编码每一点的结果为：如果点位于盒子表面或内部则为白色，否则为黑色，如图 13-26（b）所示。事实上，比较图 13-26（b）和图 13-24（h），可明显看到在 RGB 向量空间分割会产生准确得多的结果。其实从一定意义上说，在原彩色图像中，分割目标的颜色与定义的"微红色"更接近。


(a)


(b)

图 13-26　RGB 空间分割

（a）由封闭的矩形所示的感兴趣的彩色原像。

（b）与图 13-24（h）相比，RGB 向量空间分割的结果。

13.5.3　彩色图像的边缘检测

边缘检测是图像分割的一个重要工具，在这一节中，将对以分量图像为基础计算边缘和在彩色空间直接计算边缘这两个边缘检测方法进行比较。

前面介绍过梯度边缘检测算子，但梯度对向量并没有定义。这使人们很自然的想到分别计算各分量图像的梯度，然后再将图像合成。然而，这样合成的图像会导致错误的结果，下面通过一个例子来说明产生错误的原因。

考虑两幅 M×M 彩色图像（M 为奇数）（图 13-27（d）和（h）），它们分别由图 13-27（a）～（c）和图 13-27（e）～（g）中的三个分量图像合成。

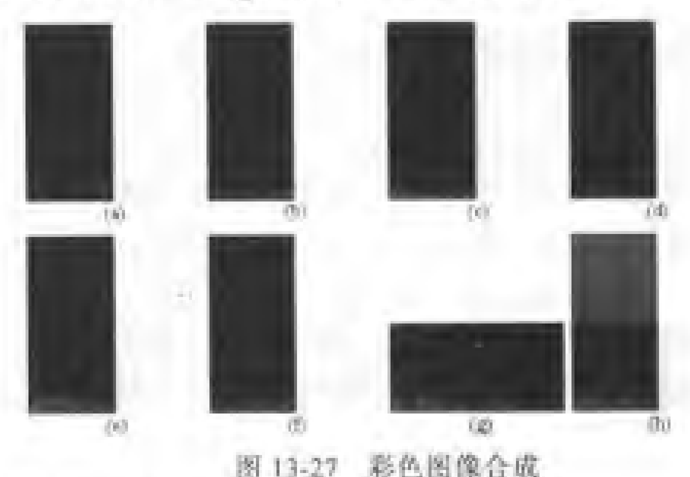


图 13-27　彩色图像合成

（a）～（c）R、G、B 分量图像。

（d）产生的彩色图像。

（f）～（g）R、G、B 分量图像。

（h）产生的彩色图像。

例如，按下式计算每一个分量图像的梯度：

$$\nabla f = mag(\nabla \vec{f}) = [G_x^2 + G_y^2]^{\frac{1}{2}} = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{\frac{1}{2}}$$

其中，$\nabla \vec{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$。将得到的各分量的梯度结果相加形成两幅相应的 RGB 梯度图像，在点 $[(M+1)/2,(M+1)/2]$ 处，两种情况下得到的梯度值是相同的。而直观地看，图 13-27（d）中图像该点处的梯度更强，因为 R、G、B 图像的边缘在该图像中处在相同的方向上，而图 13-27（h）的图像只有两个边缘在相同的方向上。可以看到，分别处理三个独立平面，然后相加合成梯度图像会导致错误结果。如果考虑的问题是只检测三个分量中的一个边缘，则单独分量方法常可得到可接受的结果。然而，如果要求更高的准确度，则很明显需要一个新的可用于向量梯度计算的定义。下面是 Di Zenzo［1986］基于此提出的一种方法。

先定义向量 $\vec{c}$（$\vec{c}(x,y) = \begin{bmatrix} c_R(x,y) \\ c_G(x,y) \\ c_B(x,y) \end{bmatrix} = \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix}$）在任意点 (x,y) 处的梯度（幅度和方向）。对标量函数 $f(x,y)$ 来说，梯度是在坐标 (x,y) 处指向 f 的最大变化率方向的向量，可以将梯度的这个概念对向量扩展。令 $\vec{r}$、$\vec{g}$、$\vec{b}$ 是 RGB 彩色空间沿 R、G、B 轴的单位向量，则可定义向量为：

$$\vec{u} = \frac{\partial R}{\partial x}\vec{r} + \frac{\partial G}{\partial x}\vec{g} + \frac{\partial B}{\partial x}\vec{b} \quad (13.5.3)$$

$$\vec{v} = \frac{\partial R}{\partial y}\vec{r} + \frac{\partial G}{\partial y}\vec{g} + \frac{\partial B}{\partial y}\vec{b} \quad (13.5.4)$$

数量 g_{xx}，g_{yy} 和 g_{xy} 定义为这些向量的点乘，如下所示：

$$g_{xx} = \vec{u}\cdot\vec{u} = \vec{u}^{T}\vec{u} = \left|\frac{\partial R}{\partial x}\right|^2 + \left|\frac{\partial G}{\partial x}\right|^2 + \left|\frac{\partial B}{\partial x}\right|^2 \quad (13.5.5)$$

$$g_{yy} = \vec{v}\cdot\vec{v} = \vec{v}^{T}\vec{v} = \left|\frac{\partial R}{\partial y}\right|^2 + \left|\frac{\partial G}{\partial y}\right|^2 + \left|\frac{\partial B}{\partial y}\right|^2 \quad (13.5.6)$$

$$g_{xy} = \vec{u}\cdot\vec{v} = \vec{u}^{T}\vec{v} = \frac{\partial R}{\partial x}\frac{\partial R}{\partial y} + \frac{\partial G}{\partial x}\frac{\partial G}{\partial y} + \frac{\partial B}{\partial x}\frac{\partial B}{\partial y} \quad (13.5.7)$$

R、G、B 及由此得到的 g 是 x 和 y 的函数。利用该表示法，$\vec{c}(x,y)$ 的最大变化率方向可以由角度给出：

$$\theta = \frac{1}{2}\arctan\left[\frac{2g_{xy}}{(g_{xx} - g_{yy})}\right] \quad (13.5.8)$$

(x,y) 点在 θ 方向上变化率的值由下式给出：

$$F(\theta) = \left\{\frac{1}{2}\left[(g_{xx} + g_{yy}) + (g_{xx} - g_{yy})\cos 2\theta + 2g_{xy}\sin 2\theta\right]\right\}^{\frac{1}{2}} \quad (13.5.9)$$

因为 $\tan(\alpha) = \tan(\alpha \pm \pi)$，如果 θ_0 是式（13.5.9）的一个解，则 $\theta_0 \pm \pi/2$ 也是。因此可以

得到，$F(\theta)=F(\theta+\pi)$，F 仅需对 θ 值在半开区间 $[0,\pi)$ 计算。式（13.5.8）提供两个相隔 90^0 的值，这意味着该方程涉及每一点 (x,y) 的正交方向，沿着这些方向之一，F 最大，沿其他方向其值最小。

【例 13-9】 在向量空间的边缘检测。

图 13-28（b）是图 13-28（a）图像的梯度，它是用上面讨论的向量方法得到的。图 13-28（c）显示了先计算每一个 RGB 分量图像的梯度，然后在每一坐标点 (x,y) 处叠加相应的 3 个分量得到的图像。向量梯度图像的边缘细节图 13-28（b）比图 13-28（c）中单独平面梯度图像的细节更完全，例如，研究对象的右眼部周围的细节。图 13-28（d）图像显示了在每一点 (x,y) 处的两种梯度图像间的差别。

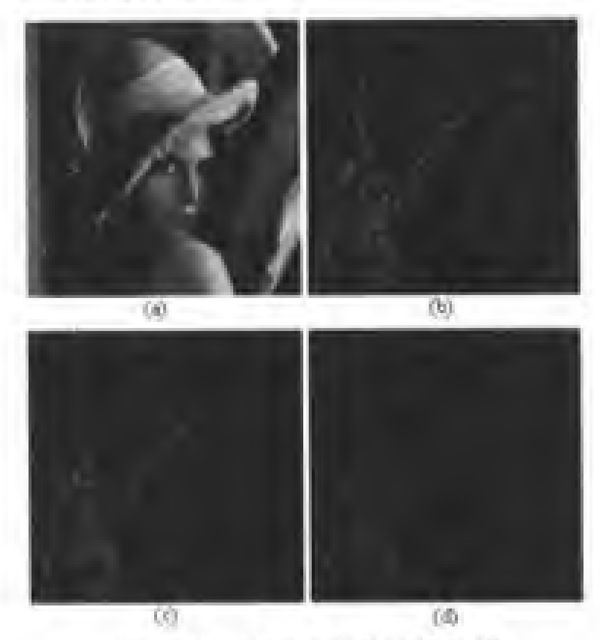


图 13-28 向量空间的边缘检测

（a）RGB 图像。

（b）在 RGB 彩色空间计算的梯度。

（c）在分量图像基础上计算然后相加的梯度。

（d）（b）和（c）之间的差别。

注意：两种方法得到的可接受的结果。图 13-28（b）中额外的细节是否值得附加计算量（与 Sobel 算子的执行相反，它用于产生单独平面的梯度）仅由给定问题的要求决定。

图 13-29 显示了三个分量梯度图像，在相加和标定时可用来得到图 13-28（c）。

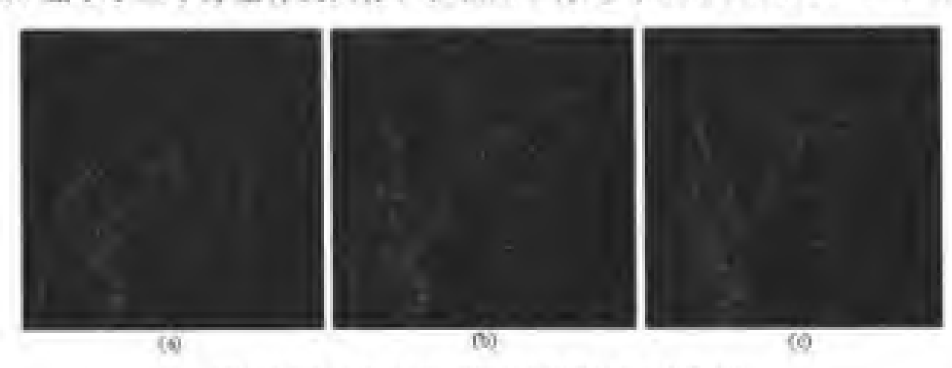


图 13-29 图 13-28 彩色图像的分量梯度图

（a）红分量。

（b）绿分量。

（c）蓝分量。

13.6 伪彩色和假彩色处理

伪彩色和假彩色图像处理是图像处理中的两项很实用的技术。伪彩色处理（Pseudocoloring）是将一幅图像的每个灰度级匹配到彩色空间中一点的方法，即将一幅单色图像用某种方法映射到一幅彩色图像的过程；假彩色处理（Falsecoloring）是将一幅彩色图像或多光谱图像映射到RGB空间中新位置上的过程。

13.6.1 伪彩色处理

对于一幅单色图像进行伪彩色处理的有利之处就在于使肉眼能辨别出比单色图像更多的、经过着色后的物体细节部分。一般来说，在单色显示器上，肉眼只能辨识256个灰度级中的二十几种灰度，经过伪彩色处理后，就能看到更多的色调，提高肉眼对图像中一些细节的分辨能力。

设$f(x,y)$为一幅单色或黑白图像，$R(x,y)$、$G(x,y)$、$B(x,y)$为$f(x,y)$映射到RGB空间的三个对应的分量值，则伪彩色处理可以表达为：

$$R(x,y)=F_R[f(x,y)] \tag{13.6.1}$$

$$G(x,y)=F_G[f(x,y)] \tag{13.6.2}$$

$$B(x,y)=F_B[f(x,y)] \tag{13.6.3}$$

其中F_R、F_G、F_B为某种伪彩色处理中的映射函数。例如，将一幅256个灰度级的多灰度图像处理成一幅按可见光谱中彩色光的波长来表示其灰度值分布的伪彩色图像，可以用如下所示的映射函数来实现：

$$R(x,y)=\begin{cases}0 & , \quad 0\leqslant f<63\\ 0 & , \quad 64\leqslant f<127\\ 4f(x,y)-510 & , \quad 128\leqslant f<191\\ 255 & , \quad 192\leqslant f\leqslant 255\end{cases} \tag{13.6.4}$$

$$G(x,y)=\begin{cases}254-4f(x,y) & , \quad 0\leqslant f<63\\ 4f(x,y)-254 & , \quad 64\leqslant f<127\\ 255 & , \quad 128\leqslant f<191\\ 1022-4f(x,y) & , \quad 192\leqslant f\leqslant 255\end{cases} \tag{13.6.5}$$

$$B(x,y)=\begin{cases}255 & , \quad 0\leqslant f<63\\ 510-4f(x,y) & , \quad 64\leqslant f<127\\ 0 & , \quad 128\leqslant f<191\\ 0 & , \quad 192\leqslant f\leqslant 255\end{cases} \tag{13.6.6}$$

经过上式的伪彩色处理，原黑白图像中的一些细节被赋予了一定的颜色，提高了人眼对这些细节的辨别能力，达到了图像增强的目的。如果与边缘检测方法相结合，可以给某边缘添加一种颜色（如红色），就能鲜明的突出某一物体的边缘，为图像检测或识别提供了方便。

13.6.2 假彩色处理

假彩色处理是将一幅彩色图像中的每一个像素的 RGB 值重新映射到 RGB 色空间中的一个新点的过程。假彩色处理是日常生活中经常碰到的一个操作过程，例如调节彩色电视机的色调、饱和度的过程实际上就是假彩色处理。又如红光成像设备拍摄了 N 幅不同波段上的 N 幅图像：$f_1(x,y)$、$f_2(x,y)$，…，$f_N(x,y)$，可以将它们经过假彩色处理再现出可见光谱图像，其处理函数如下：

$$R(x,y)=F_R[f_1(x,y),f_2(x,y),\dots,f_N(x,y)] \quad (13.6.7)$$

$$R(x,y)=F_R[f_1(x,y),f_2(x,y),\dots,f_N(x,y)] \quad (13.6.8)$$

$$R(x,y)=F_B[f_1(x,y),f_2(x,y),\dots,f_N(x,y)] \quad (13.6.9)$$

其中 F_R、F_G、F_B 为映射函数；$R(x,y)$、$G(x,y)$、$B(x,y)$ 为显示空间三基色分量。

注意： 伪彩色或假彩色处理不改变图像像素的几何位置，只改变其颜色，所以这两种技术也可以归于图像增强技术，可以与肉眼视觉特性相结合设计它们的映射函数 F_R、F_G、F_B 来达到增强人们需要的目标图像的目的。

13.7 彩色图像的噪声

可以将前面提到的噪声模型用于彩色图像。通常，彩色图像噪声的内容在每个彩色通道中具有相同的特性，但是，同样的噪声可能对不同彩色通道造成的影响不同，其中一种可能是个别通道的电子学故障，然而，更可能是由用于每个彩色通道的相对照射强度的差异造成的。

例如，CCD 摄像机使用红滤色器将减小用于红传感器的照射强度，CCD 传感器在低照明情况下就是噪声源，因此，这种情况将导致 RGB 图像红色分量比其他两个分量更容易引入噪声。

【例 13-10】 把 RGB 图像噪声转换为 HSI 的效果说明。

在这个例子中，将简单看一下彩色图像中的噪声及当从一个彩色模型转换为另一个彩色模型时如何转移噪声。图 13-30（a）–（c）显示了由高斯噪声退化的一幅 RGB 图像的三个彩色平面，图 13-30（d）是合成的 RGB 图像。


(a)


(b)


(c)


(d)

图 13-30 RGB 图像的噪声转换为 HSI 的效果

注意： 细网纹噪声在彩色图像中相比单色图像不太容易引起注意。

图 13-31（a）–图 13-31（c）显示把图 13-30（d）中的 RGB 图像转换为 HSI 图像的结果，用原图像的 HSI 分量（图 13-31）比较这些结果，并注意如何有效地减小噪声图像色调与饱和度分量，这分别是由式（13.2.13）和式（13.2.14）求余弦与取最小值操作的非

线性造成的，另一方面，图 13-31（c）中的强度分量比三个有噪声的 RGB 分量图像中的任一个都稍微平滑一些，这是因为：强度图像是 RGB 图像的平均，如式（13.2.15）所指出的那样。

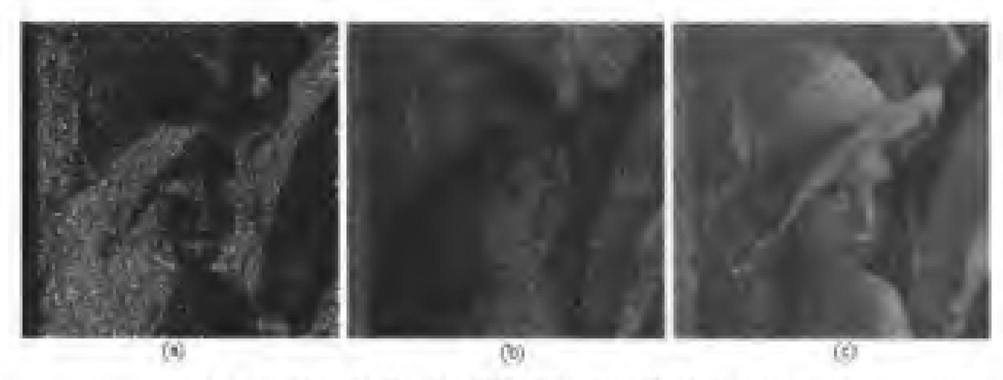


图 13-31　图 13-30（d）含噪声的彩色图像的 HSI 分量

（a）色调。

（b）饱和度。

（c）强度。

在这种情况下，仅在 RGB 通道受噪声影响并转换到 HSI 空间后，才把噪声散布到所有 HSI 分量图像上。

图 13-32（a）为一幅 RGB 图像，其中绿色分量被椒盐噪声污染，椒盐噪声的概率为 0.05，图 13-32（b）～（d）的 HSI 分量图像清楚的显示噪声怎样从绿的 RGB 通道散布到所有的 HSI 图像上，因为 HSI 分量的计算用到 RGB 所有的分量。



图 13-32　RGB 图像转换到 HIS 时的噪声散布现象

（a）绿平面受椒盐噪声污染的 RGB 图像。

（b）HSI 图像的色调图像。

（c）饱和度分量。

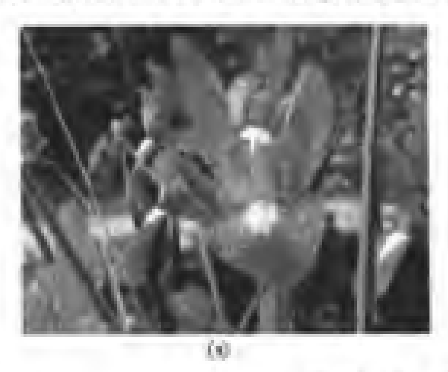
（d）强度分量。

13.8 彩色图像的压缩

因为描述彩色图像要求的比特数比描述灰度的数量大3到4倍，所以图像数据压缩在存储和传输彩色图像中起到核心作用。对于前边章节的RGB，CMY（K）和HSI图像，作为压缩对象的数据是各彩色像素的分量（例如，在RGB图像中像素的红、绿、蓝分量）；它们是传递彩色信息的方法。压缩是对图像进行处理，以减小冗余度或消除不相干数据。下面简要说明一下彩色图像压缩这一概念。

【例13-11】 彩色图像压缩实例。

图13-33（a）显示了一幅蝴蝶花的24bit RGB全彩色图像。图像中对红、绿、蓝分量的表示都使用了8bit。图13-33（b）是从（a）中图像的一种压缩方案中重建的图像，事实上该图像是压缩然后解压缩后原图像的一个近似。虽然压缩后的图像不能直接显示，在输入到彩色监视器之前必须解压缩，但压缩图像对原图像的每230数据bit仅包含1数据bit（即1个存储bit）。假定压缩后的图像可通过Internet传输1分钟，那么原图像的传输至少要4小时。当然，传输后的数据必须解压缩后才能观看，但解压缩在几秒钟就可以做完。该例中引入的产生图13-33（b）的标准是JPEG 2000压缩算法。


(a)

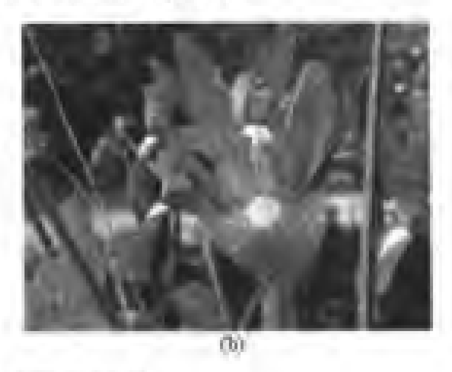

(b)

图13-33　彩色图像压缩

（a）原RGB图像。

（b）图像（a）的压缩解压缩结果。

注意： 重建的近似图像有点模糊，这是许多有损压缩技术的特点，可以通过改变压缩程度加以减小或消除。

小结

本章主要介绍了各种彩色模型及彩色坐标转换，包括RGB、HSI等彩色模型；同时简述了各种灰度处理技术在彩色图像中的扩展应用，包括图像平滑和锐化、分割和边缘检测。最后介绍了彩色图像的压缩。

重点： 各种彩色模型，特别是RGB彩色模型、HSI彩色模型及二者之间的转换，以及对彩色图像在各彩色空间进行的各种变换。

难点： 对彩色图像在RGB或HSI彩色空间进行分割、边缘检测、平滑、锐化等处理是本章的难点。

疑点： 有的彩色图像处理过程可直接沿用灰度图像处理的方法，而有的则不可以，需注意区分；伪彩色与假彩色是两个不同的概念，伪彩色处理的原始图像是一幅单色图像，而假彩色处理的原始图像是一幅彩色图像。

练习十三

一、选择题

1. 品红色（Magenta）是由（　）两种颜色合成。
   A. 红（Red）+绿（Green）　　B. 红（Red）+蓝（Blue）
   C. 红（Red）+黄（Yellow）　　D. 绿（Green）+蓝（Blue）
2. 下列（　）是常用的彩色模型。
   A. RGB　　B. XYZ　　C. UCS　　D. HSI
3. 一幅彩色图像或多光谱图像映射到 RGB 空间中新位置上的过程是（　）。
   A. 伪彩色处理　　B. 假彩色处理

二、思考题

1. 在简单的 RGB 图像中，R、G、B 分量图像有个分量水平强度分布曲线示意图如图 13-34 所示，在该图像的最左边、中间、最右边我们分别可以看到什么颜色？

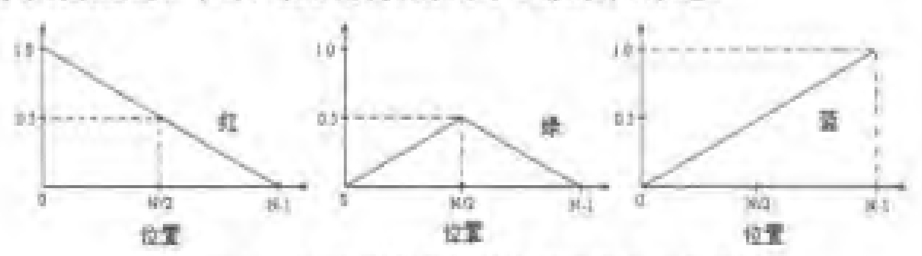


图 13-34　RGB 图像各分量水平强度分布曲线图

2. 在彩色 RGB 系统中，每一幅 RGB 分量图像是一幅 8 比特图像，共有多少不同的灰度？
3. 写出 RGB 到 HSI 的的彩色变换公式，其中，RGB 各分量分别用 R、G、B 表示，HIS 各分量分别用 H、S、I 来表示。
4. 根据图 13-16 写出蓝色、品红色、红色的补色分别是什么？
5. 图 13-35 为图 13-8 中表示的 216 种安全 RGB 色阵列，用其行、列号标注每一元素，左上角的元素是（1，1），最右边底部的元素为（12，18），试写出下列颜色的标注：

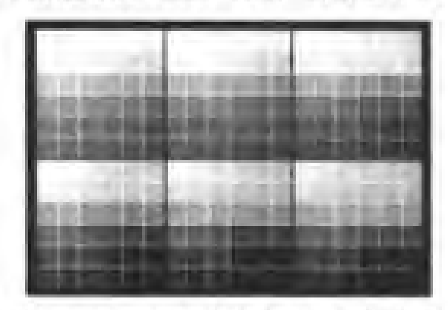


图 13-35　216 种安全 RGB 色阵列

（1）最纯的绿色。
（2）最纯的蓝色。

（3）最纯的红色。

三、上机题

1. 试编程实现用 5×5 平均模板对图 13-36 进行平滑，对 RGB 每一分量进行平滑并显示，对 HSI 图像的强度分量进行平滑后转化为 RGB 图像并显示，观察二者差别。



图 13-36 图像

2. 对题 1 中的图分别在 HSI 空间和 RGB 空间进行分割并显示分割结果，观察结果差别。

第 14 章　模式识别初步

模式识别诞生于 20 世纪 20 年代，随着 40 年代计算机的出现，50 年代人工智能的兴起，模式识别在 60 年代初迅速发展成为一门学科。它所研究的理论和方法在很多科学和技术领域中得到了广泛的重视，推动了人工智能系统的发展，扩大了计算机应用的可能性。几十年来，模式识别研究取得了大量的成果，在很多地方得到了成功的应用。但是，由于模式识别涉及到很多复杂的问题，现有的理论和方法对于解决这些问题还有很多不足之处。

本章主要内容有以下几个方面：

（1）模式识别和模式的概念。

（2）模式识别系统。

（3）关于模式识别的一些基本问题。

（4）模式识别方法简介。

14.1　模式识别和模式的概念

人们在生活中时时刻刻都在进行模式识别。环顾四周，人们能够认出周围的物体是桌子、椅子，能够认出对面的人是张三、李四；听到声音，人们能区分出是火车长鸣还是蜜蜂嗡嗡，是猫叫还是狗吠；能够区分是谁在说话，说什么内容；闻到气味我们能知道是臭豆腐还是红烧肉等等。人们所具备的这些模式识别的能力看起来极其平常，谁也不会感到惊讶，就连猫和狗都能分辨它们的主人，更低等的动物也能区分食物和敌害。因此过去的心理学家也没有注意到模式识别的能力是个值得研究的问题，就像苹果落地一样见惯不惊。只有在计算机出现之后，当人们企图用计算机来实现人或动物所具备的模式识别的能力时，它的难度才逐步为人们所认识。

本书讨论的模式识别是指用计算机实现人的模式识别能力。由于目前计算机的模式识别在多数方面还远不如人，因此研究人脑中的模式识别过程对提高机器的能力是有益的；反之，研究及其模式识别的能力对于理解人脑中的过程也有很大的帮助，认知心理学的很多新模型即得益于此。

什么是模式呢？广义上来说，存在于空间和时间中可观察的事物，如果人们可以区分它们是否相同或是否相似，都可以称之为模式。但模式所指的不是事物本身，而是人们从事物获得的信息。因此，模式往往表现为具有时间或空间分布的信息。由于本书主要讨论的是用计算机进行模式识别，信息进入计算机之前通常要经过取样和量化，在计算机中具有时空分布的信息表现为向量即数组。数组中元素的序号可以对应时间与空间，也可以对应其他的标识，例如，医生根据各项化验指标判断疾病种类的模式识别过程中，各种化验项目并不对应实际的时间或空间。因此，对于上面所说的时间与空间应作更广义、更抽象的理解。

人们为了掌握客观事物，按事物相似的程度组成类别。模式识别的作用和目的就在于面对某一具体事物时将其正确地归入某一类别。例如，数字“4”可以有各种不同的字体或写法，但它们都属于同一类，即使人们看到从未见过的某种写法的“4”，也能正确地将其

分到“4”这一类别中去。从不同的角度看同一张人脸，视网膜上的成像也不同，但人们可以识别出这个人是谁，把所有不同角度的像都归入某个人这一类。如果给每个类命名，并且用特殊的符号来表达这个名字，那么模式识别可以看成是从具有时间和空间分布的信息向着符号作的映射。

通常，把通过对具体的个别事物进行观测所得到的具有时间和空间分布的信息称为模式，而把模式所属的类别或同一类中模式的总体称为模式类（或简称类）。也有人习惯于把模式类称为模式，而把个别具体的模式称为样本，这种用词的不同可以从上下文弄清其含义，并不会引起误解。

14.2 模式识别系统

有两种基本的模式识别的方法，即统计模式识别方法和结构（句法）模式识别方法，与此相应的模式识别系统都由两个过程所组成，即设计和实现。设计是指用一定数量的样本（叫作训练集或学习集）进行分类器的设计。实现是指用所设计的分类器对待识别的样本进行分类决策。本书只讨论统计模式识别方法。

基于统计方法的模式识别系统主要由四个部分组成：数据获取，预处理，特征提取和选择，分类决策，如图 14-1 所示。

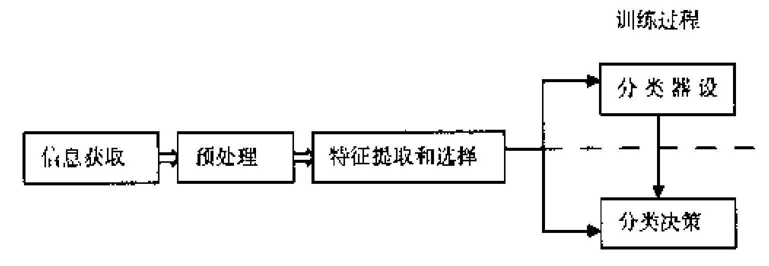


图 14-1 模式识别系统的基本构成

下面简单地对这几个部分作些说明。

1. 数据获取

为了使计算机能够对各种现象进行分类识别，要用计算机可以运算的符号来表示所研究的对象。通常输入对象的信息有下列三种类型，即：

（1）二维图像。如文字、指纹、地图、照片这类对象。

（2）一维波形。如脑电图、心电图、机械震动波形等。

（3）物理参量和逻辑量。前者如在疾病诊断中病人的体温及各种化验数据等；后者对某参量正常与否的判断或对症状有无的描述，如疼与不疼，可用逻辑值即 0 和 1 表示。在引入模糊逻辑的系统中，这些值还可以包括模糊逻辑值，比如：很大、大、比较大等。

通过测量、采样和量化，可以用矩阵或向量表示二维图像或一维波形。这就是数据获取的过程。

2. 预处理

预处理的目的是去除噪声，加强有用的信息，并对输入测量仪器或其他因素所造成的退化现象进行复原。

3. 特征提取和选择

由图像或波形所获得的数据量是相当大的。例如，一个文字图像可以有几千个数据，

一个心电图波形也可以有几千个数据，一个卫星遥感图像的数据量就更大。为了有效地实现分类识别，就要对原始数据进行变换，得到最能反映分类本质的特征。这就是特征提取和选择的过程。

一般把原始数据组成的空间叫做测量空间，把分类识别赖以进行的空间叫做特征空间，通过变换，可把维数高的测量空间表示的模式变为在维数较低的特征空间中表示的模式。在特征空间中的一个模式通常也叫做一个样本，它往往可以表示为一个向量，即特征空间中的一个点。

4. 分类决策

分类决策就是在特征空间中用统计的方法把被识别的对象归为某一类别。基本做法是在样本训练集基础上确定某个判决规则，使按这种判决规则对被识别对象进行分类所造成的错误识别率最小或引起的损失最小。

14.3 关于模式识别的一些基本问题

本节讨论与模式识别有关的一些基础性问题，由于这些问题多数还没有满意的解决方法，所以不能在正文中讨论；但这些困难的问题对于模式识别来说又很重要，因此，在此作简单的说明对于更深入地理解和进一步发展模式识别地理论技术是有益的。

1. 模式类的紧致性

为了能在某个空间中进行分类，通常假设同一类的各个模式在该空间中组成一个紧致集。从这个紧致集中的任何一点可以均匀地过渡到同一集合中的另外一点，而在过渡途中的所有各点都仍然属于这个集合。

一般来说，在 D 维空间中要用超表面进行分类，假若用平面图来表示 D 维空间中点的分布，通常有如图 14-2 所示的三种情况，其中（a）没有临界点，（b）有许多临界点，（c）的临界点已经多到使分类不可能实现。

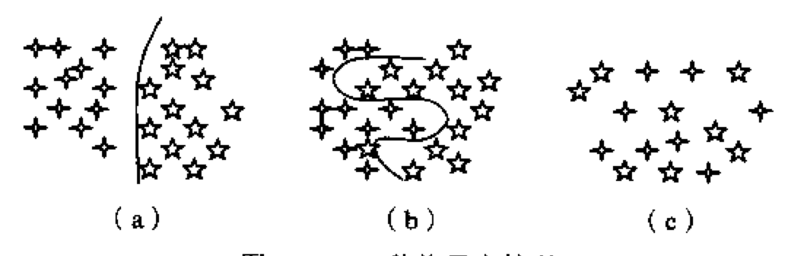


图 14-2 三种临界点情况

从上面的讨论中可以看到，紧致集具有以下性质：

（1）临界点的数量与总的点数相比很少。

（2）集合中任意两个内点可以用光滑线连接，在该连接线上的点也属于这个集合。

（3）每个内点都有一个足够大的邻域，在该邻域中只包含同一集合中的点。

假若每个模式类都满足紧致性假设，则解决模式识别问题就不会碰到原则上的困难。但对于很多实际问题这个假设是不成立的。例如，任何一个分类问题，至少在传感器（例如摄像机）输出的测量空间或其他原始描述上就完全不满足紧致性要求。另一方面，假若所给的识别任务已经很好解决，就可以设计出一种变换方法，使测量空间上属于同一类的所有各点都映射到特征空间上的同一点，而把另一类的所有各点都映射到特征空间上的另

外一点，且使这两点相隔一个显著的距离。显然，在这个特征空间中，模式类是满足紧致性假设的。

因此可以说，只要模式类是可分的，其中就存在这样一个空间，使变换到这个空间中的集合是满足紧致性要求的。这样一种变换只能在解决识别任务的过程中来求取，是和具体问题紧密相关的，现在还没有统一的解决这种变换的有效理论和方法。

2. 相似与分类

模式识别是把具体实物归入某一类别的过程。要进行归类，首先要有类存在。通常设计模式识别系统时，菲勒标准是认为地冲系统外给定的，通过设计或有监督的学习过程使系统能完成特定的识别任务。这种方法的优点是能设计出较经济的系统，软件规模不会过于庞大；缺点就是系统学习能力不强，使设计开发工作变得极为繁重。

如果用集合论中的子集和元素来表示模式类和模式，可以用集合论中的概念讨论相似关系。在一个集合 M 中可以定义一个关系 R，如果对所有的 $x \in M, xRx$ 成立，则称关系是满足对称和自返的；如果对于 $x, y \in M, xRy, yRz \Rightarrow xRz$，则称关系 R 是传递的。同时满足自返、对称和传递关系成为等价关系。

例如相等就是一种等价关系。满足等价关系的集合必定可以划分为若干子集，即 $M = \bigcup_i M_i$ 且 $M_i \cap M_j = \varnothing (i \neq j)$。在同一子集 M_i（或称等价类）中的各个元素在一定意义上是不可区分的。如果把一个子集当成一个模式类，则满足等价关系的各类间有明确的界限，或者说是可区分的。但相似关系不具有传递性。例如父亲与儿子相似，儿子与母亲相似，但是父亲并不一定与母亲相似。因此实际的相似与分类问题远不像集合表达那样简单。

目前得到广泛应用的相似性度量是在空间中定义的某种距离。给定一个输入样本集合 δ，用 D 维空间中的一个点表示某个样本，两个样本 x_j 和 x_k 之间的相似性度量 $\delta(x_j, x_k)$ 应该满足以下要求：

（1）相似性度量应该为非负，即 $\delta(x_j, x_k) \geqslant 0$。

（2）样本本身之间相似性度量应该为最大。

（3）相似性度量应满足对称性，即 $\delta(x_j, x_k) = \delta(x_k, x_j)$。

（4）在模式类满足紧致性条件下，相似性应该是点间距离的单调函数。

在各种空间中，只要定义一种距离度量，就可以用这种距离度量的非增函数作为相似性度量。例如，在 D 维欧几里德空间，可以选择某个随距离增加而下降的函数 f 作为相似性度量，即：

$$\delta(x_k, x_j) = f\left(\sqrt{\sum_{i=1}^{D}(x_{ki} - x_{ji})^2}\right)$$

在有些情况下，可以采用两个向量之间的夹角来度量相似性，例如：

$$\delta(x_k, x_j) = \cos^{-1}\frac{x_k^T x_j}{\|x_k\| * \|x_j\|}$$

以上式子中，x_k、x_j 是表示模式 k 和模式 j 的向量，x_{ki}、x_{ji} 是它们的第 I 个分量。

在讨论使用距离度量之前，需要解决两个更深层次的问题：

（1）怎样决定分类？

（2）怎样才能满足紧致性条件？

分类的复杂性，在于不存在客观的分类标准，因为任何分类都是带有主观性的。例如，鲸在生物学中属于哺乳类，应该和牛算作一类动物；但是从产业的角度看，捕鲸属于水产业，鲸和鱼是一类，而牛属于畜牧业，所以和鲸不同类。虽然科学的传统是追求客观性的，但模式识别本来是大脑中的功能，其主观性是不可避免的。从这里得到的启发是，分类问题不单纯是数学问题，还需要有生物学的观点。

3. 特征的生成

特征是决定相似性与分类的关键。当分类的目的决定之后，如何找到合适的特征就成为认知与识别的核心问题。从输入的原始信息得到特征通常需要复杂的非线性运算，直接找这种算法是非常困难的。但是由于认知与识别是人脑的基本功能，人们可以用仿生学的观点，从生物的信息处理过程中寻求启发。分析动物对感觉信息处理的过程就可以知道，在动物和人的脑中存在着对特定特征起反应的神经元，而且形成由简单到复杂逐层提取特征的结构。

为了设计一个能够通过学习生成新特征的系统，这个系统必须拥有足够丰富的特征作为基础。这些特征也可以通过相同的学习方法逐层建立起来，但最终总有一些底层的特征是需要事先给定的。为了分别讨论特征不同的生成方式，可以把特征粗略地分为底层、中层、高层三个层次。这样三个层次的划分是为了讨论与学习有关的特点人为分开的，实际上每个大层中包含有很多层，大层之间也没有明确清晰的界线。

14.4 模式识别简介

14.4.1 简介

1. 模式

所有能够被感受到或耳闻目睹的事物，如果能够区分它们，都可以称之为模式。模式不是指事物的本身，而是指人们从事物获取的信息。人每天都在做大量模式识别的工作。

模式识别分二类：{ 结构（句法）模式识别；统计模式识别（神经网络模式识别）

2. 模式识别过程的三个阶段

对于一幅含有多个物体的数字图像，模式识别过程由三个阶段组成：

图像分割 ⟹ 特征抽取 ⟹ 分类

3. 特征提取

要建立模式识别系统，首先要测量对象的特性，产生描述参数，这些参数称为特征。一个对象的所有参数值组成向量，称为特征向量。

良好特征的四个特点：

（1）可区别性（不同类）。

（2）可靠性（同类）。

（3）独立性（特征间）。

（4）数量小（复杂性）。

4. 分类器的设计

设计分类器的目的就是建立分类器的逻辑结构和分类规则。

1）逻辑结构

逻辑结构，如图 14-3 所示。

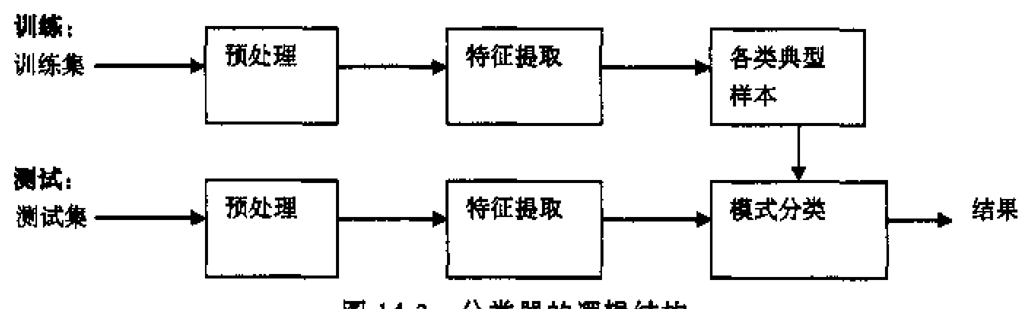


图 14-3　分类器的逻辑结构

关键环节：

（1）提出一种特征提取方法，并给出相应的预处理。

（2）建立每类的典型样本。

（3）度量测试样本与典型样本的相似程度。

2）最小距离判别规则

如果有 N 个类别 $\omega_1,\omega_2,\ldots,\omega_N$，分别用样本集 $A_1,A_2,\ldots,A_N$ 表示，则对未知样本 x，可用以下规则来分类：

$$D(x,A_i)=\min_{j=1,2,\cdots,N} D(x,A_j)\Rightarrow x\in\omega_i$$

其中，$D(x,A_j)$ 是 x 到 A_j 的某种距离度量。大多数分类规则都可以转换成阈值规则。

14.4.2　特征选择

特征抽取的一般原则：

先测试一组直觉上合理的特征，然后将其减少成数目合适的最佳集。

典型方法：

统计分类方法，统计决策理论（Bayes 理论）。

K－L 变换方法（PCA），Fisher 方法。

1.　K-L 变换

在 n 维随机向量空间中，每个样本仅为 n 维空间中一个点。对于一类对象组成的样本集，它们分布在 n 维空间的一个超平面上，通过 K－L 变换可以找到超平面的坐标基。将样本的特征用新坐标基表达，可获得无冗余的表达，如图 14-4 所示。

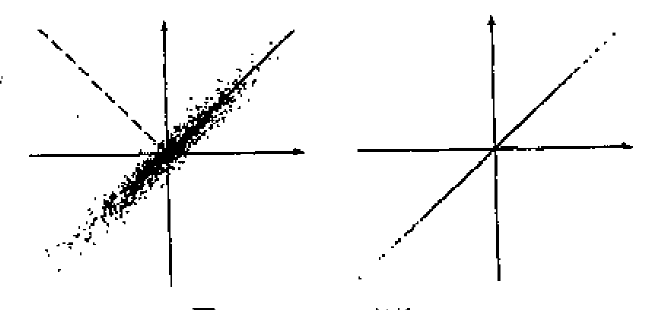


图 14-4　K-L 变换

2.　Fisher 变换

假设一个集合包含 N 类样本，一个好的特征抽取方法应该是：同类特征相聚紧密，不

同类特征分离尽可能远。

Fisher 变换可以找到这样一个特征子空间，将样本投影到该空间，可以获得最佳鉴别特征，如图 14-5 所示。

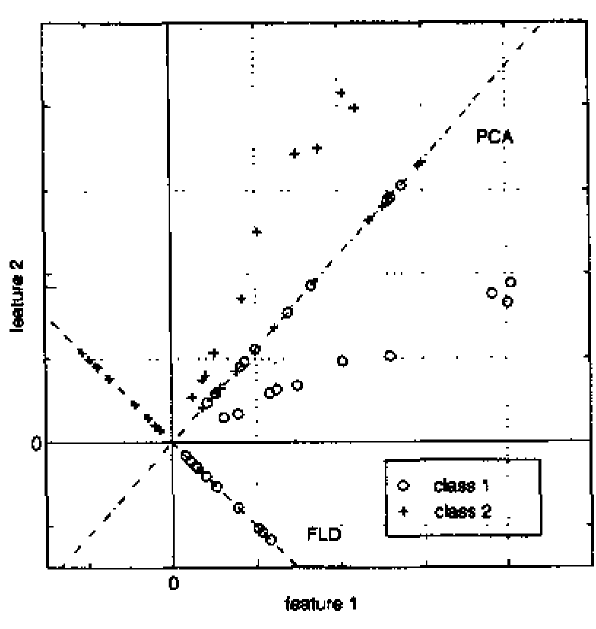


图 14-5　Fisher 变换

3. K-L 变换在人脸识别上的应用

（1）图像归一化。

每张人像用双眼对齐。

（2）采用直方图修正法使图像具有统一的均值和方差。

（3）用一训练集作训练。

一张 $d \times d$ 维人脸像，构成 d^2 维向量：

$X = (b_{11}b_{12},\ldots,b_{1d}b_{21}b_{22},\ldots,b_{2d}b_{d1}b_{d2},\ldots,b_{dd})^t$

一张人脸是 d^2 维空间一点，所有人脸都有五官，相似，样本值处于一个超平面（或人脸的像素间有很大的相关性）。

通过 K－L 变换找超平面的坐标基。

设训练样本集中有 M 幅人脸 $\{x_i, i = 0,1,\ldots,M-1\}$，其协方差矩阵为：

$\mathrm{E} = E\{(\mathbf{x}-\overline{\mathrm{X}})(\mathbf{x}-\overline{\mathrm{X}})^t\}$

或：

$$E = \frac{1}{M}\sum_{i=0}^{M-1}(x_i - \overline{\mathrm{X}})(x_i - \overline{\mathrm{X}})^t$$

其中：x_i 为第 i 个训练样本的图像向量，$\overline{\mathrm{X}} = \frac{1}{M}\sum_{i=0}^{M-1} x_i$。

E 是 $d^2 \times d^2$ 维的对称矩阵，半正定的，进行特征分析：

$Ev = \lambda v$

注意： 直接计算特征向量，特征根困难！

事实上，$E = \frac{1}{M}\sum_{i=0}^{M-1}(x_i - \overline{X})(x_i - \overline{X})^t = \frac{1}{M}XX^T$，　$d^2 \times d^2$

其中：$X = [x_0 - \bar{X}, x_1 - \bar{X}, ..., x_{M-1} - \bar{X}]$

故，构造矩阵：

$E' = X^T X \in R^{M \times M} \qquad M << d^2$

求出特征根 λ_i 和对应的特征向量 $v_i (i = 0,1,2,...,M-1)$。

由 SVD 定理：E 的特征向量 u_i 为：

$$u_i = \frac{1}{\sqrt{\lambda_i}} X v_i \qquad i = 0,1,2,...M-1$$

（4）特征选择。

令：

$\lambda_0 \geqslant \lambda_1 \geqslant ... \geqslant \lambda_{r-1}$

$u_0, u_1, ..., u_{r-1}$ 重排序。

选择前 k<=r 个特征向量，如果：

$$\frac{\sum_{i=0}^{k-1} \lambda_i}{\sum_{i=0}^{r-1} \lambda_i} \geqslant \alpha \qquad (\text{例如} \alpha = 95\%)$$

则，$u_0, u_1, ..., u_{k-1}$ 为人像表达坐标基。

（5）人像表达。

由 k 个特征向量做正交基，建立的子空间为像特征空间。u_i 称为特征脸，即：

$u = [u_0, u_1, u_2, ..., u_{k-1}]$

对 $\forall$ 人像 f，向“特征脸”子空间投影，可获得投影的系数向量：

$y = u^T f$

即为 f 的特征表达。

（6）识别测试。

选择一个相似性度量 $d(x, y)$。

制作每个人的标准样本 A_i。

对 $\forall f$，求出 y，计算 y 与 A_i 的距离 $d(y, A_i)$。

若：

$d(y, A_i) = \min_{j=1,2,...N} d(y, A_j)$，则$y \in \omega_i$。

小结

模式识别学科还在不断发展，人工神经网络和各种新的最优化技术在模式识别中广泛的应用，以及作为统计模式识别基础的统计学习理论的新发展都预示着模式识别的前景无可限量。本章只是给读者一个对模式识别的初步印象。主要介绍了模式识别和模式的概念、模式识别系统、关于模式识别的一些基本问题、模式识别简介等内容。有关更加深人的知识，需要读者参考模式识别方面的书籍，数字图像处理是模式识别的前提条件，所以首先应学好图像处理这个基础知识，才能学好更加深人的学科。

重点：本章重点在于了解模式识别的概念，理解什么是模式识别。

难点： 本章难点在于如何理解模式分类的重要意义。

疑点： 模式识别的研究方向是什么。

练习十四

一、选择题

1. 一个计算机模式识别系统基本上是由（　　）三部分组成的。

   A. 数据采集　　B. 数据处理　　C. 数据存储　　D. 分类决策

2. 模式识别分为（　　）两类。

   A. 结构模式识别　　B. 统计模式识别

   C. 分类模式识别　　D. 自动模式识别

二、思考题

1. 为什么图像处理技术是模式识别的基础。
2. 什么是模式识别，简述其概念。
3. 模式识别的研究主要集中在哪两个方面？

三、上机题

设计一个人脸识别的程序。